НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

«На правах рукопису»
УДК 004.049

«До захисту допущено»
Завідувач кафедри
_____ І.Р. Пархомей
(підпис)
“___”_____2019 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 126 «Інформаційні системи та технології»

на тему: Підвищення безпеки системи навчання робототехніці _____

Виконав (-ла): студент (-ка) другого курсу, групи ІК-81мп_
(шифр групи)

_____Нікітін Валерій Андрійович_____ _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник доцент, _____к.т.н., доцент, Крилов Є. В. __ _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант ____НК____ __к.т.н., доцент, Пасько В. П._____ _____
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали) (підпис)

Рецензент _____к.т.н., доцент, Катін П. Ю._____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2019 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

Рівень вищої освіти – другий (магістерський)

Спеціальність 126 «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ І.Р. Пархомей
(підпис)
«___»_____2019 р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Нікітіну Валерію Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема дисертації Підвищення безпеки системи навчання робототехніці ____,

науковий керівник дисертації Крилов Є. В., к.т.н., доцент _____ ,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «_28_»_жовтня__ 2019 р. №3770-с_

2. Термін подання студентом дисертації 9 грудня 2019 року _____

3. Об'єкт дослідження Вразливості та методи забезпечення безпеки веб-орієнтованих системи _____

4. Предмет дослідження Функції фреймворку Laravel, що використовуються для створення веб-орієнтованих систем _____

5. Перелік завдань, які потрібно розробити Виконати аналіз існуючих загроз та способів попередження атак на веб-орієнтовані системи, провести діагностику існуючої системи, розробити підсистему для підвищення захисту з урахуванням певних обмежень, дослідити ефективність та працездатність створеної підсистеми _____
_____

6. Орієнтовний перелік ілюстративного матеріалу – шість плакатів _____

7. Орієнтовний перелік публікацій Нікітін В. А., к.т.н. Крилов Є. В., ст. в. Анікін В. К. Аналіз атак та захисту Web-додаток з використанням Cross Site Scripting вразливості//XV International scientific and practical Conference “Scientific horizons – 2019”; Нікітін В. А., к.т.н. Крилов Є. В., ст. в. Анікін В. К. Аналіз сучасних Web-вразливостей//XV International scientific and practical Conference “Scientific horizons – 2019”

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
НК	Пасько В. П., доцент		
Перевірка на співпадіння	Лісовиченко О. І., доцент		

9. Дата видачі завдання 26 вересня 2018 року

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Формування проблематики	02.09.2019 – 08.09.2019	
2	Аналіз проблематики	09.09.2019 – 15.09.2019	
3	Постановка задачі	16.09.2019 – 23.09.2019	
4	Розробка основних модулів веб-орієнтованої системи	24.09.2019 – 30.09.2019	
5	Оптимізація безпеки веб-орієнтованої системи	01.10.2019 – 07.10.2019	
6	Розробка підсистеми	08.10.2019 – 13.10.2019	
7	Тестування та покращення підсистеми	14.10.2019 – 20.10.2019	
8	Впровадження підсистеми	21.10.2019 - 27.10.2019	

Студент _____ В. А. Нікітін
(підпис) (ініціали, прізвище)

Науковий керівник дисертації _____ Є. В. Крилов
(підпис) (ініціали, прізвище)

РЕФЕРАТ

Магістерська дисертація: 83 с., 25 рис., 27 табл., 2 додатки та 17 джерел.

Об’єктом дослідження є вразливості та методи забезпечення безпеки веб-орієнтованих систем навчання робототехніці.

Метою даної роботи є підвищення безпеки системи навчання робототехніці.

У ході роботи розглянуто основні вразливості веб-орієнтованих систем, мережеві атаки, фактори, які впливають на безпеку програмного забезпечення. Проведено аналіз існуючих способів для забезпечення інформаційної безпеки веб-орієнтованих систем.

Результатом роботи є підсистема, що підвищує захист веб-орієнтованої системи на основі фреймворку Laravel, з використанням запропонованого способу пошуку найоптимальнішого вектору оптимізуючих перетворень.

Ключові слова: безпека веб-орієнтованих систем, мережева безпека, фреймворк laravel, вразливості програмного забезпечення, вектор оптимізуючих перетворень.

ABSTRACT

The master’s thesis: 83 p., 25 fig., 27 tabl., 2 appendices and 17 sources.

The theme of this thesis is “Increasing safety robotics training system”.

The subject of the study is vulnerabilities and security methods for web-based robotics training systems.

The purpose of this work is to increase the safety of the robotics training system.

In the course of the work the main vulnerabilities of web-oriented systems, network attacks, factors that affect the security of the software were considered. An analysis was made of existing ways to ensure information security of web-oriented systems.

The result is a subsystem that enhances the protection of Laravel's web-based system against certain threats.

Keywords: web oriented system security, network security, laravel framework, software vulnerabilities, vector of optimal transformations.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до магістерської дисертації

на тему: “Підвищення безпеки системи навчання робототехніці”

Київ – 2019 року

ЗМІСТ

ВСТУП

Актуальність загроз цілісності і конфіденційності інформації вимагає уважного ставлення до завдання її захисту. Завдання забезпечення безпеки інформації 20 років тому вирішувалася за допомогою засобів криптографічного захисту, встановлення міжмережевих екранів, розмежування доступу. Зараз цих технологій недостатньо, будь-яка інформація, що має фінансову, конкурентну, військову чи політичну цінність, опиняється під загрозою. Додатковим ризиком стає можливість перехоплення управління критичними об'єктами інформаційної інфраструктури.

Відомості про спроби розкрадання ресурсів інформації, що належать державі, в більшості випадків закриті. А дані про кількість злочинів в кредитно-банківській сфері та щодо громадян регулярно розкриваються кіберполіцією. Так, за перші шість місяців 2018 року кількість злочинів, вчинених у сфері інформаційних технологій, становить 4 тисячі правопорушень. Таким чином, метою роботи є підвищення безпеки веб-орієнтованої системи навчання робототехніці.

Для досягнення даної мети були поставлені та вирішені наступні завдання:

1. розробити спосіб для вибору найоптимальнішого вектору оптимізуючих перетворень;
2. виконати аналіз існуючих загроз та способів попередження атак на веб-орієнтовані системи навчання робототехніці;
3. провести діагностику існуючої системи навчання робототехніці;
4. розробити підсистему для підвищення захисту системи з урахуванням обмеження за швидкодією;
5. дослідити ефективність та працездатність створеної підсистеми.

Об'єктом дослідження є вразливості та методи забезпечення безпеки веб-орієнтованих систем навчання робототехніці.

Предметом дослідження є вибір оптимального вектору оптимізуючих перетворень для фреймворку Laravel, що використовується для створення веб-орієнтованих систем.

Запропоновано спосіб для вибору найоптимальнішого вектору оптимізуючих перетворень та алгоритми підвищення безпеки системи від Brute force, мережевої DoS атак та перехоплення конфіденційних даних користувача.

Практичним результатом є створення веб-орієнтованої системи навчання робототехніці та впроваджена підсистема для підвищення безпеки.

Робота складається з 5 розділів. У першому розділі розглядається актуальність проблеми та існуючі види загроз на програмне забезпечення. У другому розділі розглядаються способи для забезпечення безпеки програмного забезпечення та конфіденційних даних кінцевих користувачів. Третій розділ присвячено аналітичному обґрунтуванню запропонованого способу для вибору найоптимальнішого вектору оптимізуючих перетворень. У четвертому розділі приведений детальний опис розробленої системи навчання робототехніці, схеми бази даних, наведені результати діагностики системи на наявність вразливостей, блок-схеми алгоритмів та їх реалізація та вибір найоптимальнішого вектору оптимізуючих перетворень для даної системи.

П’ятий розділ присвячений комерціалізації розробки у вигляді стартап-проекту.

РОЗДІЛ 1 АКТУАЛЬНІСТЬ ПРОБЛЕМИ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ВЕБ-ОРІЄНТОВАНИХ СИСТЕМ

1.1 Поняття інформаційної безпеки

Термін "інформаційна безпека" може мати різний зміст і трактування в залежності від контексту. В загальному випадку це є ступінь захищеності інформації та інфраструктури системи від довільного або цілеспрямованого впливу природного або штучного характеру, що можуть призвести до збитків суб'єктам інформаційних відносин. До них відносяться власники та користувачі інформаціїх[1].

Поняття інформаційної безпеки було введено у ДСТУ "Захист інформації. Технічний захист інформації. Терміни та визначення". Існує три основні компоненти, які повинні бути враховані під час створення системи:

— конфіденційність – стан інформації, при якому доступ до неї здійснюють тільки суб'єкти, що мають на нього право.
— цілісність – стан інформації, при якому відсутнє будь-яка її зміна або зміна здійснюється тільки навмисно суб'єктами, що мають на нього право;
— доступність – стан інформації, при якому суб'єкти, які мають право доступу, можуть реалізовувати його безперешкодно.

Забезпечення інформаційної безпеки є складним завданням, для вирішення якої потрібний комплексний підхід.

Законодавчий рівень є основою для побудови системи захисту інформації, так як дає базові поняття предметної області і визначає міру покарання для потенційних зловмисників. Цей рівень відіграє координуючу і спрямовуючу роль і допомагає підтримувати в суспільстві негативне ставлення до людей, які порушують інформаційну безпеку[2].

1.2 Загрози інформаційній безпеці

Загрозу інформації називають потенційно можливий вплив або вплив на автоматизовану систему з подальшим нанесенням збитку власникам або користувачам.

Загрози інформаційної безпеки проявляються не самостійно, а через можливу взаємодію з найбільш слабкими ланками системи захисту, тобто через фактори вразливості. Загроза призводить до порушення діяльності систем на конкретному об'єкті-носії[3].

Основні вразливості виникають внаслідок дії наступних факторів:

— недосконаліть програмного забезпечення, апаратної платформи;
— різні характеристики архітектури автоматизованих систем в інформаційному потоці;
— частина процесів функціонування систем є неповноцінною;
— неточність протоколів обміну інформацією та інтерфейсу;
— складні умови експлуатації і розташування інформації.

Найчастіше джерела загроз запускаються з метою отримання незаконної вигоди внаслідок заподіяння шкоди інформації. Але можлива випадкова дія загроз через недостатню міру захисту і масовість дії небезпечного фактору.

За класами вразливості поділяють на:

— об'єктивні;
— випадкові;
— суб'єктивні.

Якщо усунути або як мінімум послабити вплив вразливостей, можна уникнути повноцінної загрози, спрямованої на систему зберігання інформації.

Об'єктивні вразливості безпосередньо залежать від технічної побудови обладнання на об'єкті, що вимагає захисту, і його характеристик. Повноцінне позбавлення від цих чинників неможливо, але їх часткове усунення досягається за допомогою інженерно-технічних способів:

1. пов'язані з технічними засобами випромінювання:

— електромагнітні (побічні варіанти випромінювання і сигнали від кабельних ліній, елементів технічних засобів);
— звукові (акустичні або з додаванням вібросигналів);
— електричні (прослизання сигналів в ланцюжки електричної мережі, за наведенням на лінії і провідники, по нерівномірному розподілу струму).

2. ті, що активуються:

— шкідливе програмне забезпечення, нелегальні програми, технологічні виходи з програм;
— закладки апаратури – фактори, які впроваджуються безпосередньо в телефонні лінії, в електричні мережі або просто в приміщення.

3. ті, що створюються особливостями об'єкта, який знаходиться під захистом**:**

— розташування об'єкта (видимість і відсутність контрольованої зони навколо об'єкту інформації, наявність вібро- або звуковідбивальних елементів навколо об'єкта, наявність віддалених елементів об'єкта);
— організація каналів обміну інформацією (застосування радіоканалів, оренда частот або використання загальних мереж).

4. ті, що залежать від особливостей елементів-носіїв**:**

— деталі, що володіють електроакустичними модифікаціями (трансформатори, телефонні пристрої, мікрофони і гучномовці, котушки індуктивності);
— речі, які потрапляють під вплив електромагнітного поля (носії, мікросхеми та інші елементи).

Випадкові вразливості фактори залежать від непередбачених обставин та особливостей оточення інформаційного середовища. Їх практично неможливо передбачити в інформаційному просторі, але важливо бути готовим до їх швидкого усунення[4].

Усунути такі неполадки можна за допомогою проведення інженерно-технічного розгляду:

1. збої і відмови роботи систем:

— внаслідок несправності технічних засобів на різних рівнях обробки та зберігання інформації (в тому числі і тих, що відповідають за працездатність системи і за контроль доступу до неї);
— несправності і старіння окремих елементів (розмагнічування носіїв даних, таких як дискети, кабелі, з'єднувальні лінії і мікросхеми);
— збої різного програмного забезпечення, яке підтримує всі ланки в ланцюзі зберігання і обробки інформації (антивіруси, прикладні і сервісні програми);
— перебої в роботі допоміжного обладнання інформаційних систем (неполадки на рівні електропередачі).

2. ослабляють інформаційну безпеку чинники:

— пошкодження комунікацій на зразок водопостачання або електропостачання, а також вентиляції, каналізації;
— несправності в роботі захисних пристроїв (паркани, перекриття в будинку, корпуси обладнання, де зберігається інформація).

Суб'єктивні вразливості в більшості випадків являє собою результат неправильних дій співробітників на рівні розробки систем зберігання і захисту інформації. Тому усунення таких факторів можливо за допомогою методик з використанням апаратури і програмного забезпечення:

1. неточності і грубі помилки, що порушують інформаційну безпеку:

— на етапі завантаження готового програмного забезпечення або попередньої розробки алгоритмів, а також в момент його використання (можливо під час щоденної експлуатації, під час введення даних);
— на етапі управління програмами і інформаційними системами (складності в процесі навчання роботі з системою, настройки сервісів в індивідуальному порядку, під час маніпуляцій з потоками інформації);

— під час користування технічної апаратурою (на етапі включення або виключення, експлуатації пристроїв для передачі або отримання інформації).

2. порушення роботи систем в інформаційному просторі:

— режиму захисту особистих даних (проблему створюють звільнені працівники або діючі співробітники в неробочий час, вони отримують несанкціонований доступ до системи);

— режиму збереження і захищеності (під час отримання доступу на об'єкт або до технічних пристроїв);

— під час роботи з технічними пристроями (можливі порушення в енергозбереженні або забезпеченні техніки);

— під час роботи з даними (перетворення інформації, її збереження, пошук і знищення даних).

Ранжування вразливостей дозволяє визначити критерії оцінки небезпеки виникнення загрози і вірогідність поломки або обходу захисту інформації. Показники підраховуються за допомогою застосування ранжирування.

1.3. Типи проблем в забезпеченні інформаційної безпеки

1.3.1 Зловмисне програмне забезпечення

Шкідливі програми – це одна з можливих причин завдяки якій відбувається нанесення збитків за рахунок вразливостей у інформаційній безпеці. Шкідливі програми створюються спеціально для несанкціонованого користувачем знищення, блокування, модифікації або копіювання інформації, порушення роботи комп'ютерів або комп'ютерних мереж. До зазначеної категорії належать віруси і черв'яки, троянські програми і інший інструментарій, створений для автоматизації діяльності зловмисників.

Сучасне розповсюдження шкідливих програм є для кінцевого користувача загрозою, оскільки з кожним роком відбувається покращення та створення нових способів для проникнення на обчислювальну машину. За

рахунок цього можливе некоректне функціонування підсистем, що забезпечують захист від несанкціонованих дій зловмисників.

Шкідливим програмним забезпеченням є троянські програми, мережеві хробаки, хакерські утиліти, класичні файлові віруси. Класичні файлові віруси не використовують мережеві сервіси для впровадження в інші комп'ютери. Дублікат вірусу впроваджується на віддалені комп'ютери тільки за умови, що заражений об'єкт з певних причин виявляється активізованим на чужому комп'ютері.

Також деякі віруси можуть мати в собі властивості різних видів шкідливого програмного забезпечення. Це може бути шпигунський, або навіть троянський компонент для спотворення або видалення інформації, що зберігається на дисках (наприклад, вірус CIH).

Троянські програми – це небезпечні програми, що були створені для виконання несанкціонованих дій користувачем. Вони спрямовані на зміну, створення дублікатів, блокування або навіть видалення інформації, порушення нормального функціонування комп'ютерів або комп'ютерних мереж. На відміну від мережевих хробаків, представники даного виду не можуть створювати дублікати, що мають властивість самовідтворення. Їх основна ознака це виконання несанкціонованих дій на зараженому комп'ютері.

Вразливостями називають неспроможність системи забезпечити захист від навмисного порушення цілісності у програмному забезпеченні. Вони можуть існувати внаслідок помилок розробників у вихідному коді програми, що дозволяють захопити контроль над системою, так і неправильній архітектурі, що дає можливість проникнення у систему цілком легальними, іноді навіть документально оформлених способами. Якщо в операційній системі, додатках або навіть мовах програмування існують відомі вразливості, то така система є відкритою для вірусів, якою б захищеною вона не була.

1.3.2 Атаки

Мережева атака – вплив на віддалену або локальну обчислювальну систему для захоплення контролю або навіть дестабілізації та відмові в обслуговуванні, з метою отримання конфіденційних даних користувачів.

Основними атаками даного типу є переповнення буферу, mailbombing, використання спеціалізованих програм (сніфферів, вірусів, поштових хробаків, rootkit-ів, троянських коней), IP-спуфінг, мережева розвідка, phishing-атаки, відмова в обслуговуванні (DoS- і DDoS-атаки).

Атака для переповнення буфера базується на недоліках у реалізації систем, що можуть спричинити вихід за межі пам'яті та завершити роботу додатка, а також від імені користувача розпочати виконання впровадженого бінарного коду. Небезпека приховується також у тому, що якщо додаток був запущений з привілеями адміністратора, тоді зловмисник може отримати повний контроль над даною обчислювальною системою[5].

Сніффер пакетів – це спеціальна програма для аналізу мережевого трафіку, що застосовує мережеву карту та працює в режимі, який дозволяє приймати усі пакети незалежно від адресації. Цей режим дозволяє отримати пакети по фізичним каналах, а мережевий адаптер надсилає через додаток на обробку. Слід зазначити, що сніффер перехоплює трафік, який проходить через певний домен. Загалом, дане програмне забезпечення не є шкідливим, оскільки воно застосовується для діагностики та аналізу нормального функціонування мережевого трафіку.

Але виходячи з того, що деякі додатки не використовують шифрування даних при передачі (FTP, telnet, POP3, SMTP), використовуючи сніффер стає можливим отримати конфіденційну інформацію. Це може призвести до витоку даних для аутентифікації користувачів.

Мережева розвідка – це збір та накопичення інформації про мережу з використанням загальнодоступних додатків та утиліт. Для успішної атаки на систему, зловмиснику необхідно отримати достатню кількість інформації про

технології, за допомогою яких була створена система та сервіси, які використовуються.

Розвідка проводиться у вигляді запитів DNS та сканування на наявність відкритих портів. Ці дії дозволяють дізнатися власника домену та які адреси відносяться до нього. За рахунок адреси, що була розкрита, стає можливим визначити хости, що працюють. Після цього відбувається сканування портів для складання списку сервісів, які вони надають. За цією інформацією, зловмисник робить аналіз характеристик додатків, які працюють на даних хостах та використовує для злому[6].

IP-спуфінг відбувається за рахунок того, що зловмисник видає себе за санкціонованого користувача або має доступ до системи зсередини.

Атака DoS є класичним прикладом. Вона розпочинається з адреси, що не належить зловмиснику та приховує зловмисника. Загалом, IP-спуфінг використовується для вставки помилкової інформації або спеціальних команд у звичайний потік даних, що передаються між додатками клієнта та сервера. Також можливий варіант використання каналу зв'язку між пристроями, що є одноранговими. Щоб отримати двосторонній зв'язок, зловмиснику необхідно модифікувати таблиці маршрутизації для направлення трафіку на власну IP-адресу[7].

При успішній зміні таблиці маршрутизації, зловмисник має можливість робити запити як санкціонований користувач.

SQL-ін'єкція – це вид атаки, що використовує вразливості серверної частини системи. Для проведення такого виду атак достатньо, щоб не серверній частині була відсутня фільтрація вхідних даних. В результаті запит набуває зовсім інший зміст, і в разі недостатнього захисту здатен не тільки зробити вивід конфіденційної інформації, а й змінити або видалити дані.

PHP-ін'єкція – один із способів злому веб-сайтів, що працюють на PHP. Він полягає в тому, щоб впровадити спеціально сформований зловмисний сценарій в код веб-додатку на серверній стороні сайту, що призводить до виконання довільних команд.

XSS атака – це атака, за допомогою якої можливе впровадження довільного коду у згенеровану HTML-сторінку шляхом передачі довільного значення змінним.

За рахунок створення запиту від HTML-сторінки на серверну частину, у якості скрипту передається значення змінної. Даний скрипт на запит створює HTML-сторінку, яка містить необхідні значення змінних та надсилає зловмиснику на Web-браузер.

Таким чином, XSS-атака виконується за рахунок вразливостей серверної частини на комп'ютери клієнтів. Використовуючи даний вид атаки найчастіше викрадають Cookies. Це необхідно для отримання інформації про сесії користувача, щоб перехопити керування конфіденційними даними у веб-орієнтованих системах, до тих пір, доки сервер або користувач не припинить роботу.

Зловмисник може заволодіти конфіденційною інформацією аж до отримання паролів доступу до інших веб-орієнтованих систем. Особливість даної атаки у тому, що вона наносить шкоду тільки обчислювальним машинам клієнтів, а серверна частину системи залишається у повністю робочому стані.

XPath-ін'єкція – вид вразливостей, який полягає у впровадженні XPath-виразів в оригінальний запит до бази даних XML. XPath (XML Path Language) – це мова, яка призначена для довільного звернення до частин XML документа.

XML (eXtensible Markup Language) – це всім відома мова розмітки, за допомогою якого створюються XML документи, що мають деревоподібну структуру.

Найбільш відомою формою атаки є DoS. Неможливо створити повний захист від атак даного типу. Для реалізації цього виду атаки потрібно мінімум навичок та знань, оскільки головною метою є не отримання конфіденційних даних або привілеїв адміністратора, а повний відказ в обслуговуванні користувачів даної системи.

Суть полягає у тому, щоб перевищити максимально допустимі межі працездатності системи або програми, що зробить її повністю недоступною для користування. Якщо метою є Web-сервер, то суть полягає у блокуванні вільних з'єднань, що стає перешкодою для звичайного обслуговування користувачів. Для цього можуть бути використанні стандартні Інтернет-протоколи.

1.3.3 Спам та фішинг

Спам – це електронний еквівалент паперової реклами, яку кидають у поштову скриньку. Однак спам не просто набридає і дратує. Він небезпечний, особливо якщо є частиною фішингу.

Спам у величезних кількостях розсилається по електронній пошті спамерами і кіберзлочинцями, мета яких:

- виманити гроші у деякої кількості одержувачів, що відповіли, на повідомлення;
- провести фішингову атаку, щоб обманним шляхом отримати паролі, номери кредитних карт, банківські облікові дані;
- поширити шкідливий код на комп'ютерах одержувачів.

Phishing (фішинг) – процес обману або соціальна обробка клієнтів організацій для подальшого злодійства їх ідентифікаційних даних та передачі їх конфіденційної інформації для злочинного використання.

Злочинці для свого нападу використовують спам або комп'ютери-боти. При цьому розмір компанії жертви не має значення; якість особистої інформації отриманої злочинцями в результаті нападу, має значення саме по собі.

Висновки за розділом

У даному розділі розглянуто основні види атак на веб-орієнтовані системи. Це дає можливість про загальне уявлення про види загроз та можливих наслідків, якщо своєчасно не приділити увагу та часу

інформаційній безпеці. Бездіяльність може призвести до втрати конфіденційних даних клієнтів, внаслідок чого компанія, що є власником системи, може понести великі матеріальні втрати. Але найстрашнішим наслідком є втрата довіри кінцевого користувача.

РОЗДІЛ 2 СПОСОБИ ТА ЗАСОБИ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ

Інформаційна безпека підкреслює важливість інформації в сучасному суспільстві – розуміння того, що інформація – це цінний ресурс, щось більше, ніж окремі елементи даних. Сукупність дій, які пов’язані із мінімізацією можливостей несанкціонованого доступу до системи з метою будь-якого руйнування, модифікації або затримок у доступі до інформації, називаються інформаційною безпекою.

Для забезпечення інформаційної безпеки потрібно врахувати всі можливі фактори, які можуть вплинути на цілісність, конфіденційність та доступність інформації, яка циркулює у системі.

Можна виділити наступні напрямки заходів інформаційної безпеки:

— правові;

— організаційні;

— технічні.

Правовими заходами є розроблення нормативно-правових актів, які визначають кримінальну відповідальність щодо кіберзлочинців, забезпечують авторські права розробників, впроваджують правки до цивільного та кримінального законодавства. Також відносяться контроль за розробниками програмного забезпечення та впровадження міжнародних договорів щодо обмежень, якщо вони мають негативний вплив на економічні, соціальні або навіть військові сфери країн, які укладають угоду.

Такі речі, як безпека обчислювального центру, підбір персоналу, виключення залежності працездатності системи від однієї людини, ретельно створена система захисту від дій усіх користувачів, розподілена відповідальність між працівниками, що мають забезпечити захист центру, відносяться до організаційних заходів.

Такі заходи, як безпека від несанкціонованого доступу, збереження найважливіших підсистем, перерозподіл ресурсів внаслідок непрацездатності

певних ланок, наявність протипожежного обладнання, наявність захисту від фізичного впливу, встановлення додаткових систем електроживлення, відносяться до технічних заходів.

2.1 Реалізація алгоритмів у програмному коді

Основне завдання написання захищеного коду – створення захищених програм, тобто програм, які забезпечують конфіденційність, цілісність і доступність інформації клієнта IT-компанії, а також цілісність і доступність обчислювальних ресурсів, керованих власником системи або системним адміністратором. Забезпечити виконання певних правил конкретно написання коду мало – перш за все, треба правильно організувати весь ітеративно-інкрементний процес розробки програмного забезпечення, а також забезпечити коректне управління безпекою додатків[8].

До основних принципів, що повинні бути включені у політику безпеки при роботі з додатками і при їх написанні, можна віднести:

1) наявність у штаті фахівця з інформаційної безпеки;
2) організація безперервного навчання штату;
3) класифікація помилок систем інформаційної безпеки.

Загальні принципи проектування захищених додатків:

1. ставлення до захисту системи, як невід'ємної функції створюваного програмного забезпечення;
2. на забезпечення безпеки додатку повинно бути відведено достатньо часу;
3. обов'язково повинна бути складена модель загроз: декомпозиція програмного забезпечення з виявленням властивих вразливостей; визначення рівня небезпеки та ймовірності виникнення кожної вразливості; складання матриці загроз; визначення протидій, а також дій в разі реалізації загрози;

4. визначення процедури видалення небезпечних функцій і частин в додатку;

5. повинні бути створені метрики безпеки, відповідні моделі загроз, в яких повинні бути визначені граничні пороги;

6. розробка тест-планів і періодична перевірка і контроль процесу створення програмного забезпечення відділом інформаційної безпеки на кожному етапі розробки по створеним тест-планам, по можливості, запрошеними фахівцями;

7. обов'язковий контроль безпеки модуля не тим, хто розробив цей модуль;

8. безпека повинна забезпечуватися в конфігурації за замовчуванням і при розгортанні;

9. особливий контроль за наданням прав на внесення змін до програмного забезпечення;

10. потенційна площа вразливості повинна бути якомога менше (всілякі відкриті TCP / UDP порти, що запускаються і залежні служби, динамічні веб-сторінки, частини програми або служби, що запускаються з високими привілеями);

11. повинні бути захищені всі рівні, незалежно один від одного і від інших рівнів захисту;

12. використання правила мінімальних привілеїв та грамотно складений ACL;

13. слід вести розробку з урахуванням аксіоми: зовнішні системи за замовчуванням не захищені;

14. розробити план дій на випадок збоїв або відмов;

15. не потрібно будувати систему безпеки при обмеженій інформації про створюване програмне забезпечення;

16. чітке розділення коду та даних, виключення будь-якої суміші даних і JS- або SQL-коду;

17. виправляючи помилки в захисті, потрібно перевіряти всю систему – всі модулі, намагаючись знайти там ті ж самі проблеми.

2.2 Брандмауер

Брандмауери аналізують комунікації і відхиляють пакети, які заборонено пересилати. Також вони скидають небажаний мережевий трафік і пропускають легальний трафік, який ґрунтується на створених правилах, наприклад, авторизований загальний доступ до файлів і папок. На кожному комп'ютері, починаючи з операційної системи Windows Vista, брандмауер Windows в режимі підвищеної безпеки працює в якості індивідуального брандмауера, що дозволяє здійснювати істотний захист від несанкціонованого проникнення через периметр брандмауера організації.

Мережевий трафік – це кількість інформації, що представляє собою пакет або потік пакетів, які відправляються з вихідного порту одного комп'ютера на цільовий порт іншого комп'ютера. У свою чергу, портом називається ідентифікований номером системний ресурс, що виділяється з додатком, що виконується на деякому мережевому хості, для зв'язку з додатками, виконуваними на інших мережевих хостах. Один порт може одночасно прослуховувати лише одна програма, яка повинна бути зареєстрована з зареєстрованим під свою роботу портом. Після того як вхідний пакет надходить на локальний комп'ютер, операційна система визначає номер порту призначення, після чого передає вміст пакета програмі, яка зареєструвала цей порт. Порт може перебувати в діапазоні від 1 до 65 535.

Принцип роботи індивідуального брандмауера наступний: брандмауер в режимі підвищеної безпеки перевіряє вихідні і цільові номера портів, самі порти і адреси комп'ютерів, після чого зіставляє їх із заданими правилами.

За рахунок налаштованих правил фільтрації пакетів, брандмауери відфільтровують пакети шляхом відповідності заголовків визначеним критеріям. Фільтр працює асиметрично, тобто маючи різну обробку вхідних

та вихідних пакетів, що дозволяє гнучко налаштовувати фільтрацію мережевого трафіку.

За рахунок впровадженого у брандмауері маршрутизатору, відбувається витягання заголовків з пакетів для протоколів TCP, IP, UDP та виконується їх синтаксична перевірка та аналіз. Після цього, налаштовані правила фільтрації застосовуються до пакетів в тому порядку, в якому вони знаходяться у ACL списку.

2.3 Протоколи для захищеного обміну даними

У мережі Internet об'єднано безліч комп'ютерів різних типів. Ці комп'ютери можуть використовувати різні операційні системи, але всі вони повинні підтримувати прийнятий для обміну даними в Internet стандарт реалізований на базі стека протоколів TCP / IP. Стек протоколів – розділений на рівний набір протоколів, які працюють спільно, реалізуючи певну комунікаційну архітектуру.

Зазвичай завдання того чи іншого рівня реалізуються одним або декількома протоколами. Набір мережевих протоколів, що використовуються у технології інтернет, називаються стеком протоколів TCP / IP. Існує кілька рівнів і завжди протоколи високого рівня базуються на протоколах нижчого рівня. Нижніми є протоколи фізичного і канального рівнів. Наприклад, протокол Ethernet, що описує передачу даних витою парою.

2.3.1 HTTPS

HTTPS (Hypertext Transport Protocol Secure) – це протокол, який забезпечує безпеку і конфіденційність при обміні інформацією між веб-орієнтованою системою і пристроєм користувача. Відвідувачі веб-ресурсу розраховують, що зазначені ними дані не потраплять в руки шахраїв.

Використання HTTPS обумовлена тим, що недоторканність інформації забезпечується за допомогою протоколу TLS (Transport Layer Security – безпека на транспортному рівні), який передбачає три основні рівні захисту:

1. шифрування переданих даних, щоб уникнути їх перехоплення;
2. збереження даних;
3. аутентифікація.

2.3.2 IPSec

Віртуальна приватна мережа (VPN) забезпечує безпечний тунель через загальнодоступну і, отже, небезпечну мережу. Як відомо, VPN найчастіше використовується, надаючи користувачам доступ до електронної пошти, документів, принтерів і системам з їх домашньої мережі. Безпека таких даних критично важлива. IPsec, скорочення IP Security, являє собою набір протоколів, стандартів і алгоритмів для захисту трафіку по ненадійній мережі.

IPsec підтримується практично всіма маршрутизаторами і дозволяє захистити дані в мережі. Цей протокол надає служби безпеки на рівні IP, дозволяючи системі обирати необхідні протоколи безпеки, визначати алгоритми, що будуть використовуватись для служб, і вводити будь-які криптографічні ключі, необхідні для надання потрібних послуг. Цей протокол можна використовувати для забезпечення захисту інформації, що передаються по одному або декількох каналах хостів, шлюзами безпеки або навіть між хостом та шлюзом безпеки[9].

Даний протокол забезпечує:

- конфіденційність, що запобігає крадіжці даних, використовуючи шифрування;
- цілісність гарантує, що дані не будуть змінені або замінені, використовуючи алгоритм хешування;
- аутентифікація, яка підтверджує особистість відправки даних хоста, використовуючи попередньо розділені ключі або центр сертифікації;
- захист від повторного використання, що запобігає дублюванню зашифрованих пакетів, підписуючи унікальний порядковий номер.

2.3.3 SSL/TLS

SSL – це криптографічний протокол, що використовує асиметричну криптографію для аутентифікації ключів обміну, симетричне шифрування для збереження конфіденційності, коди аутентифікації повідомлень для цілісності повідомлень. Криптографічний протокол TLS є послідовником SSL і забезпечує більшу безпеку транспортного рівня за рахунок використання псевдовипадкової функції, що розбиває вхідні повідомлення на дві частини і кожна з них оброблюється різною хеш-функцією. SSL є більш ранньої системою, TLS з'явився пізніше і він заснований на специфікації SSL 3.0, розробленої компанією Netscape Communications[10].

2.3.4 PGP

Pretty Good Privacy, він же PGP – це криптографічна програма, яка дозволяє зашифрувати інформацію таким чином, щоб ніхто сторонній не міг ні прочитати, ні змінити дані. По суті, це надійний спосіб передачі файлів, що гарантує повну конфіденційність. Це дозволить приховати кожну одиницю інформації.

Необхідність використовувати симетричний алгоритм шифрування обумовлена його високою швидкістю та можливістю отримання вихідного повідомлення.

При дешифруванні повідомлень все відбувається навпаки. Програма одержувача використовує приватний ключ для отримання сесійного ключа, а після отримання його, розшифровує вихідне повідомлення.

2.4 Цифровий підпис

Електронний підпис – це реквізит документа, що дозволяє підтвердити відповідність її власнику, а також зафіксувати стан інформації, щоб зафіксувати наявність, або відсутність змін в електронному документі з моменту його підписання.

Її застосування не допускається у випадках, пов'язаних з державною таємницею. Можливості електронного підпису фізичними особами забезпечує віддалену взаємодію з державними, навчальними, медичними та іншими інформаційними системами через інтернет. Юридичним особам електронний підпис дає допуск до участі в електронних торгах, дозволяє організувати юридично-значимий електронний документообіг і здачу електронної звітності до контролюючих органів влади. Можливості, які надає електронний цифровий підпис користувачам, зробили її важливою складовою повсякденного життя і пересічних громадян, і представників компаній[11].

2.5 Проксі-сервер

Найчастіше проксі-сервер використовується для контролювання доступу в інтернет співробітників певної корпоративної мережі. Як приклад, можна блокувати соціальні мережі, веб-орієнтовані системи, які не є необхідними для працівника.

Проксі-сервери можуть підмінити інформацію стосовно використовуваного браузеру, приховати власну IP-адресу, заблокувати використання кукі взагалі. Таким чином, при використанні проксі-сервера, сторонні джерела в інтернеті можуть отримати набагато менше інформації про користувачів, ніж при використанні прямого підключення до інтернету.

Налаштування проксі-серверу у бізнес-мережі дозволяє заблокувати доступ до ненадійних веб-орієнтованих систем, шифрувати особисту інформацію користувачів, що підвищить безпеку системи загалом.

2.6 Хмарні технології

Центр обробки даних являє собою сукупність серверів, розміщених на одному майданчику з метою підвищення ефективності і захищеності. Захист центрів обробки даних являє собою мережевий і фізичний захист, а також відмовостійкість і надійне електроживлення. В даний час на ринку

представлений широкий спектр рішень для захисту серверів і центрів обробки даних від різних загроз.

Поява віртуалізації стало актуальною причиною масштабної міграції більшості систем на хмарні технології, проте рішення задач забезпечення безпеки, пов'язаних з експлуатацією додатків в новому середовищі, вимагає особливого підходу. Багато типів загроз достатньо вивчені і для них розроблені засоби захисту, проте їх ще потрібно адаптувати для використання у хмарі[12].

В основі забезпечення фізичної безпеки лежить строгий контроль фізичного доступу до серверів і мережевої інфраструктури. На відміну від фізичної безпеки, мережева безпека в першу чергу є побудова надійної моделі загроз, що включає в себе захист від вторгнень і міжмережевий екран. Використання брандмауера в якості фільтра, з метою розмежувати внутрішні мережі центру на підмережі з різним рівнем довіри. Це можуть бути окремо сервери, доступні з Інтернету або сервери з внутрішніх мереж[13].

Висновки за розділом

У даному розділі були розглянуті методи та способи захисту веб-орієнтованих систем від несанкціонованого доступу та отримання конфіденційних даних кінцевих користувачів зловмисниками. Комплексний захист інформаційної системи забезпечується лише при розумній комбінації розглянутих методів, але найважливішу роль відіграє добре продумана архітектура програмного забезпечення, що забезпечує легке масштабування, підтримку та легке впровадження нових підсистем.

РОЗДІЛ 3 АНАЛІТИЧНЕ ОБГРУНТУВАННЯ ЗАПРОПОНОВАНОГО СПОСОБУ ВИБОРУ ОПТИМАЛЬНОГО ВЕКТОРУ ОПТИМІЗУЮЧИХ ПЕРЕТВОРЕНЬ

3.1 Нечіткі множини та їх використання для прийняття рішень оптимізації

Нечітка множина – це сукупність довільних елементів, відносно яких неможливо сказати, що ці елементи володіють певною властивістю, яке використовується для задання даної множини.

Вона відрізняється від звичайної множини тим, що неможливо остаточно сказати стосовно елементів x, які знаходяться в множині X, стосовно наявності властивості R. Можна сказати, що нечітка підмножина A універсальної множини X визначається як множина впорядкованих пар $A = \mu A x / x$, де $\mu A x$ – є характеристичною функцією належності, що приймає значення з впорядкованої множини $M = 0; 1$. Ця функція вказує ступінь належності елемента x підмножині X. Множину M називають також множиною належностей. Якщо $M = 0; 1$, то нечітка множина може розглядатися як звичайна чітка множина. Ступінь належності $\mu A x$ є суб'єктивною мірою ступеня елементу $x \in X$, відповідає поняттю, зміст якого формується нечіткою множиною A[14].

Носієм нечіткої множини A є звичайна підмножина $S A$ універсальної множини X з властивістю $\mu A x > 0$, тобто $S A = x / x \in X \wedge \mu A x > 0$.

Одним з основних методів вирішення задач оптимізації є підхід Беллмана-Заде, в якій множину альтернатив та мету прийняття рішень розглядають як рівноважні нечіткі підмножини деякої універсальної множини альтернатив. Це дозволяє звести рішення задачі до відносно простого вигляду[15].

Вимоги до вирішення задачі формалізуються наступним чином. Певна альтернатива x забезпечує досягнення мети зі ступенем $\mu_G(x)$ та задовольняє обмеження зі ступенем $\mu_C(x)$, де G – нечітка підмножина альтернатив множини

X, C – нечітка підмножина обмежень множини *X*. За таких умов нечітким рішенням *D* задачі досягнення нечіткої мети є перетин нечітких множин обмежень та мети, тобто $D = G \cap C$. Таким чином, розв'язком нечітко визначеної мети є деяка нечітка підмножина універсальної множини альтернатив *X*.

$$\mu_{G \cap C} = \begin{cases} 0, \text{ при } \mu_C(x) > T, \\ \mu_G(x), \text{ при } \mu_C(x) <= T, \end{cases} \quad (3.1)$$

де *T* – константне значення ступеня обмеження.

Якщо перетин множин визначати за правилом 3.1, тоді цільова функція розв'язку μ_D буде мати вигляд:

$$\mu_D(x) = max \{ \mu_{G1}(x, \mu_{C1}(x)), \ldots, \mu_{Gn}(x, \mu_{Cm}(x)) \}. \quad (3.2)$$

3.2 Вектори оптимізуючих перетворень та коефіцієнт захисту

Вектор оптимізуючих перетворень – це набір впроваджень, які необхідні для оптимізації певних показників системи.

В набір можуть входити функції або модулі, які виконують необхідну функціональність для підвищення безпеки системи.

Якісним показником кожного вектору є значення коефіцієнту захисту, що є ступенем досягнення мети, який розраховується за формулою 3.3:

$$P(V) = \sum_{i=0}^{n} k_i \cdot O_i, \quad (3.3)$$

де *P* – коефіцієнт захисту системи;

V – вектор оптимізуючих перетворень;

m – номер вектору оптимізуючих перетворень;

n – кількість перетворень, що входять до вектору;

i – номер поточного перетворення;

k – вірогідність атаки, для якої були реалізовані перетворення;

O – наявність захисту від атаки у векторі оптимізуючих перетворень.

Для розрахунку використовується ймовірність певної атаки, що є статистичною величиною та береться з відповідних джерел організацій, які займаються безпосередньо моніторингом кількості атак у світі. Другим значенням для розрахунку є число, яке символізує вірогідність блокування загрози, для якої була створена функція або модуль. Для спрощення, в якості значень цей показник буде приймати значення або 1, або 0, що означає повний захист, або його відсутність.

Те чи інше впровадження у будь-якому випадку несе погіршення швидкодії системі за рахунок виконання захисних операцій, таких як перевірка вхідних даних, збереження мета інформації у базах даних, запис на сервері кількості спроб авторизації клієнта з певної IP-адреси.

3.3 Розрахунок впливу векторів оптимізуючих перетворень на швидкодію системи

Для перевірки того, як вектор оптимізуючих перетворень впливає на швидкодію, можна розрахувати еталонне сумарне значення швидкості відгуку на запит певної сторінки системи та порівняти із сумарним значенням швидкості відгуку на запит до сторінок з впровадженим вектором оптимізуючих перетворень (формула 3.4).

$$T(V) = \sum_{i=0}^{n} t_i, \tag{3.4}$$

де *T* – сумарний час генерації сторінок сервером системи, ms;

V – вектор оптимізуючих перетворень;

n – кількість сторінок;

i – номер поточної сторінки;

t – час генерації сторінки сервером, ms.

Розрахувавши час кожного вектору оптимізуючих перетворень, стає можливим розрахувати приріст у часі роботи системи, тобто на скільки відсотків впала швидкодія системи (формула 3.5). Це є показником ступеня задоволення обмеженню.

$$\Delta T(V) = \frac{T(V) - T(V0)}{T(V0)} \cdot 100, \quad (3.5)$$

де $\Delta T(V)$ – приріст у часі для генерації сторінок системи в залежності від впровадженого вектору оптимізуючих перетворень, %;

$T(V)$ – час, що необхідний серверу для генерації сторінок системи з впровадженим вектором оптимізуючих перетворень, мс;

$T(V_0)$ – час, що необхідний серверу для генерації сторінок системи без впровадження вектору оптимізуючих перетворень, мс.

Після розрахунку, всі значення доцільно записати у результуючу таблицю для порівняння отриманих даних та вибору найоптимальнішого.

3.4 Створення векторів оптимізуючих перетворень та вибір оптимального

Після створення множини необхідних функцій *O*, необхідно шляхом комбінації елементів множини *O* отримати множину векторів оптимізуючих перетворень *V* (табл. 3.1).

Кожний вектор оптимізуючих перетворень складається з множини значень $M = \{0, 1\}$. Це дозволяє абстрагуватись від реалізації та позначити наявність або відсутність функцій у кожному векторі оптимізуючих перетворень.

Кількість векторів оптимізуючих перетворень відповідає максимальній кількості комбінацій можливих альтернатив.

Таблиця 3.1 – Множина альтернатив векторів
для множини оптимізуючих перетворень

Вектор оптимізуючих перетворень \ Оптимізуюче перетворення	O_1	O_2	O_3
V_0	0	0	0
V_1	1	1	1
V_2	1	0	0
V_3	0	1	0
V_4	0	0	1
V_5	1	1	0
V_6	0	1	1
V_7	1	0	1

Тоді правило перетину множин (формула 3.1) можна записати наступним чином:

$$P(V) = \begin{cases} 0, \text{ при } \Delta T(V) > \Delta T(V_0), \\ P(V), \text{ при } \Delta T(V) <= \Delta T(V_0). \end{cases} \quad (3.6)$$

Тоді, рішення можна описати наступною функцією належності використовуючи формулу 3.2:

$$P_D(V) = max \{ P_1(V_1, \Delta T(V_1)), \ldots, P_{Gn}(V_n, \Delta T(V_n)) \}. \quad (3.7)$$

Оптимальним вектором оптимізуючих перетворень буде той, значення коефіцієнту захисту який було обраним максимальним за формулою 3.3.

Висновок за розділом

У даному розділі запропоновано та обґрунтовано спосіб для оцінки захищеності веб-орієнтованої системи навчання робототехніці. Також були

розписані формули для розрахунків коефіцієнту захисту та впливу векторів оптимізуючих перетворень на швидкодію системи. Це дозволяє обрати найоптимальніший вектор оптимізуючих перетворень для підвищення безпеки інформаційної системи з використанням способу Беллмана-Заде.

РОЗДІЛ 4 РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ НАВЧАННЯ РОБОТОТЕХНІЦІ

4.1 Функціональна схема системи навчання робототехніці

Дана система має 10 модулів, які забезпечують її функціонал. На рис. 4.1 зображена схема компонентів.

Рисунок 4.1 – Схема системи навчання робототехніці

Одним з найважливіших модулів є модуль управління користувачами. За допомогою цього модулю адміністратор системи має можливість при необхідності додавати, видаляти облікові записи користувачів або редагувати вже існуючих. Користувачі в свою чергу поділяються на чотири групи: кандидат, студент, вчитель та адміністратор. Даний модуль дозволяє гнучко керувати користувачами.

Для будь-якого користувача початковою точкою для роботи із системою є модуль авторизації. Він забезпечує інтерфейс взаємодії з користувачем задля вводу поштової адреси та паролю. Після натискання кнопки для авторизації, відбувається перевірка введених даних з даними, які зберігаються у базі даних. У разі успіху користувач потрапляє до власного кабінету (рис. 4.2).

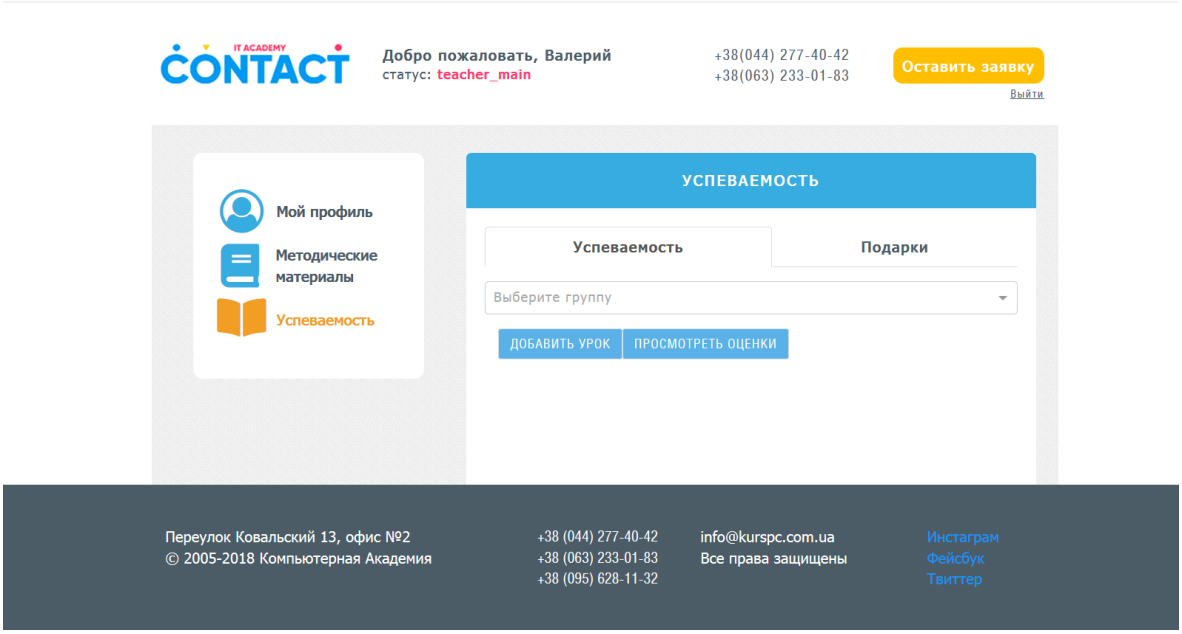


Рисунок 4.2 – Інтерфейс власного кабінету після успішної авторизації

Якщо дані для авторизації введені некоректно, тоді на стороні клієнта відображається відповідне повідомлення (рис. 4.3).

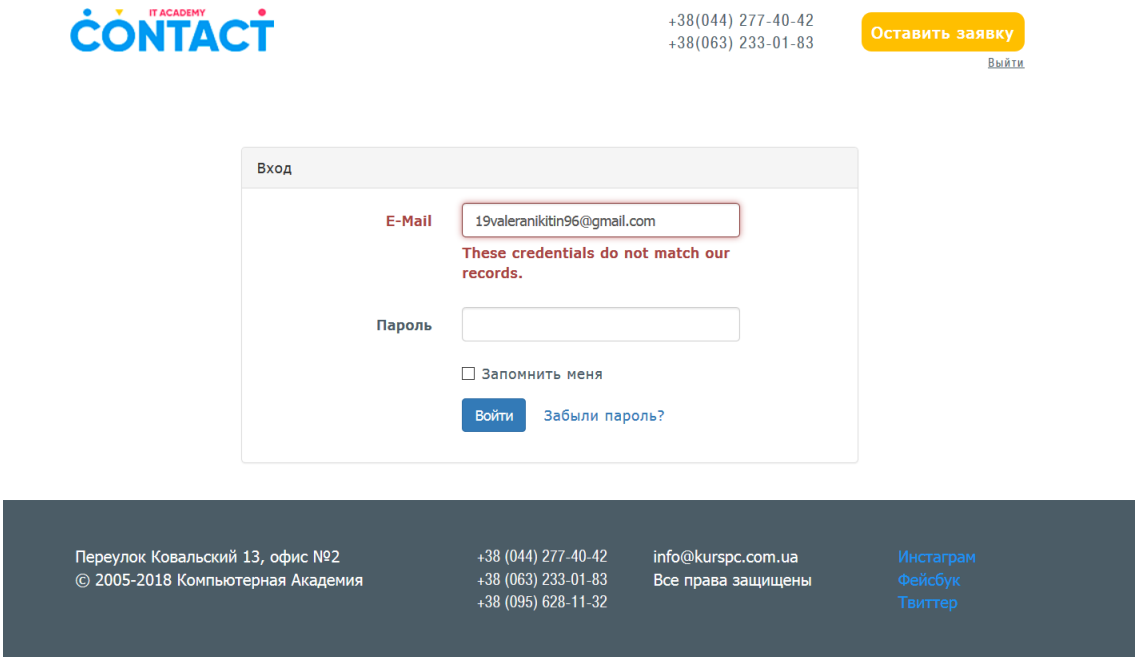


Рисунок 4.3 – Вивід повідомлення про некоректність введених даних

У разі забуття паролю, користувач має змогу відновити його вказавши адресу поштової скриньки (рис. 4.4).

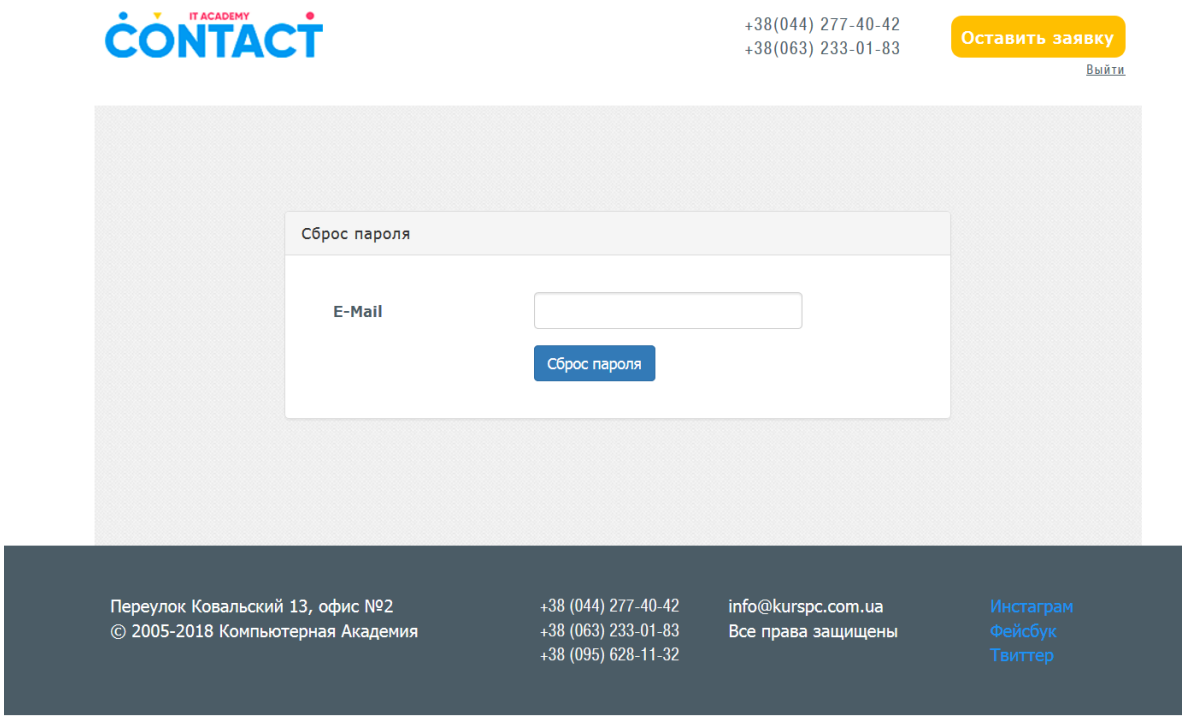


Рисунок 4.4 – Користувацький інтерфейс для відновлення паролю

Функціонал власного кабінету залежить від типу користувача, що обумовлює використання різних модулів.

Модуль формування методичних матеріалів необхідний для всіх користувачів, окрім кандидатів, оскільки завдяки йому відбувається пошук та завантаження файлів для певного заняття. Студенти можуть завантажити файли з серверу для опрацювання пройденого матеріалу та виконання домашнього завдання. Викладачі мають можливість створювати каталоги занять та завантажувати туди свої методичні матеріали та проекти-приклади (рис. 4.5). Адміністратор має той же самий функціонал для попередження форс-мажорних випадків.

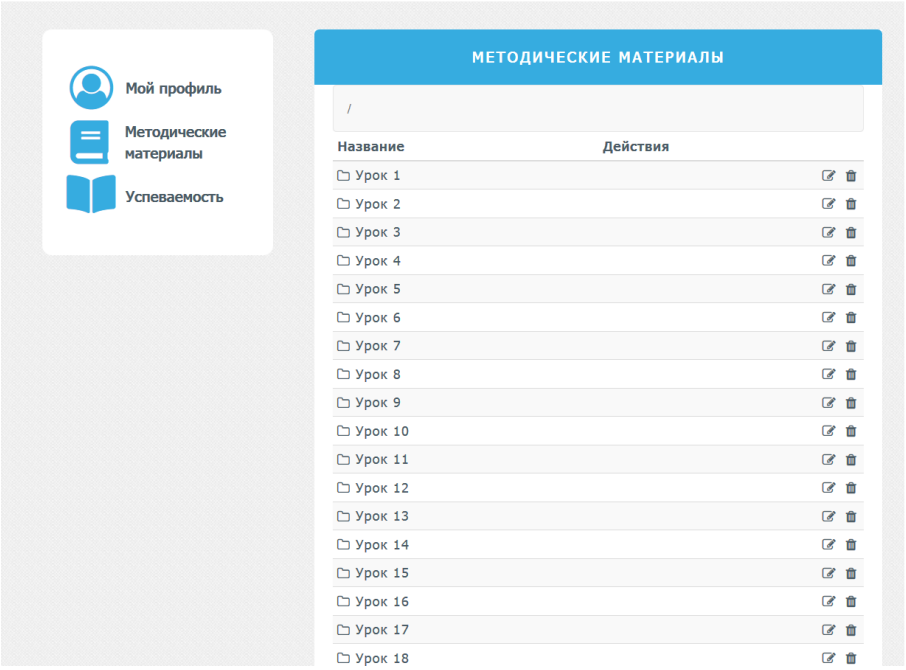


Рисунок 4.5 – Сторінка методичних матеріалів з предмету робототехніка

Модуль для управління групами дозволяє створювати, видаляти або редагувати групи, що проходить навчання за певним курсом. Для створення нової групи необхідно вказати назву, початок та кінець періоду навчання, вказати філіал, в якому будуть відбуватися заняття, аудиторія, а також напрям підготовки. Після того, як вона буде створеною, адміністратор може додавати туди вчителів та студентів. До цієї групи прив’язується певна директорія для збереження та зчитування файлів за допомогою модуля формування методичних матеріалів. Адміністратор може відслідковувати відвідування студентами занять та наявність оцінок, що проставлені вчителем (рис. 4.6 та рис. 4.7).

Модуль управління подарунками дозволяє адміністратору та студентам побачити список нагород за добре навчання. Це дозволяє стимулювати студентів добре вчитися та виконувати домашні завдання, оскільки це напряму впливає на цінність подарунка (рис. 4.8).

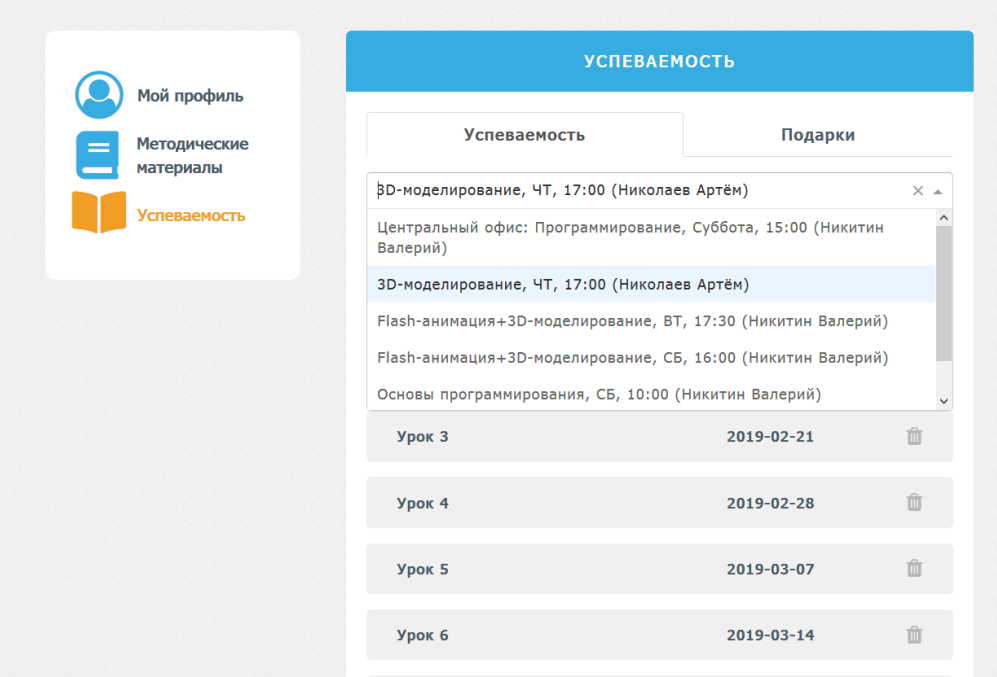


Рисунок 4.6 – Відображення груп у кабінеті адміністратора

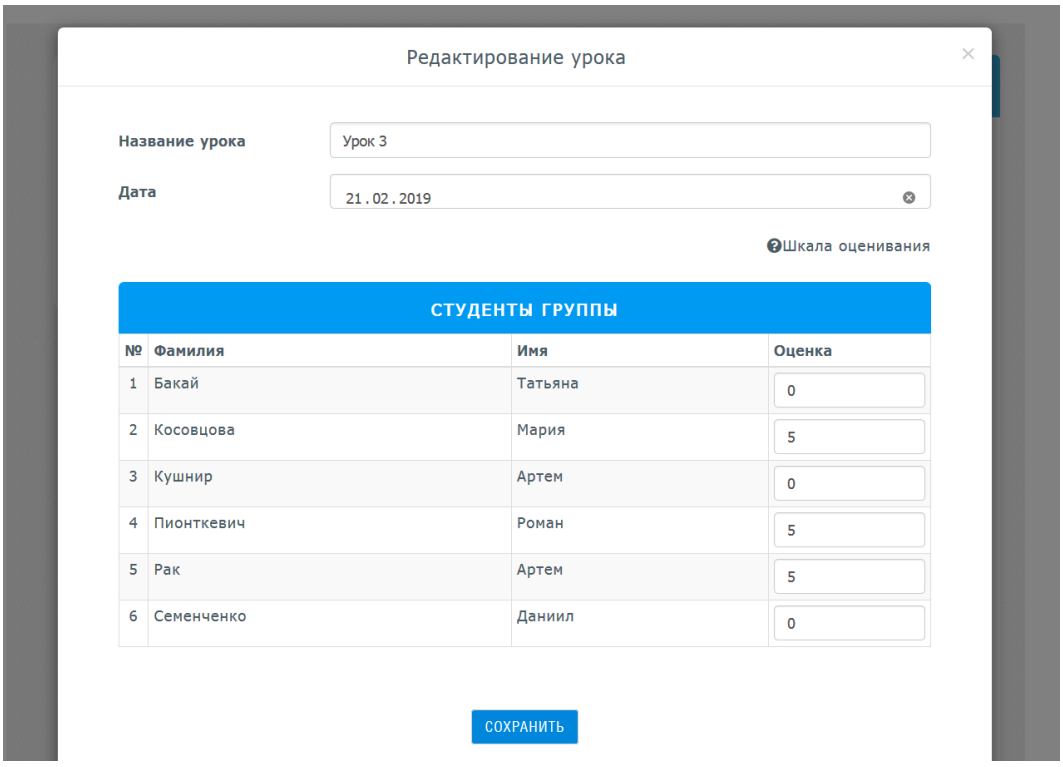


Рисунок 4.7 – Відображення успішності студентів

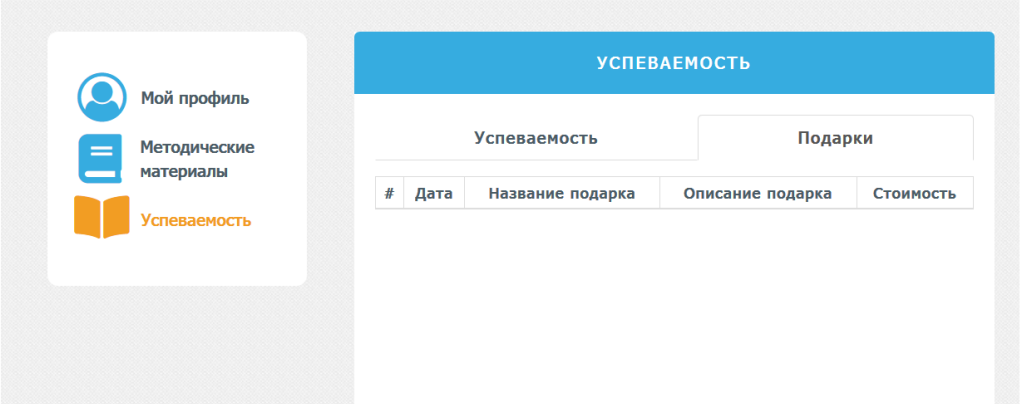


Рисунок 4.8 – Відображення подарунків за добру успішність

Модуль управління журналом успішності дозволяє студентам побачити свої оцінки, що були отримані на заняттях. Вчителі за допомогою цього функціоналу мають змогу не тільки побачити оцінки студентів, а також поставити або редагувати. Він тісно пов’язан з модулем управління подарунками, оскільки на основі цих даних формується подарунок певної цінності. За одне заняття студент може отримати від 0 до 5 балів.

Модуль розсилки повідомлень дозволяє адміністратору отримати список поштових скриньок, які відфільтровані за філіалом, напрямком курсу та групою, та зробити масову розсилку листів необхідного змісту.

Модуль формування документів необхідний для формування документа з інформацією, яку вибере адміністратор. Він може відфільтрувати за необхідними напрямками підготовки та групами, та обрати потрібні дані студентів. Далі цей документ може бути використаний для друку або створення звіту щодо успішності або відвідувань занять.

Метою модуля для формування новин є створення контенту на сайті задля донесення інформації щодо акцій або заходів до користувачів системи або гостей.

Модуль управління філіалами дозволяє адміністратору використовувати вищеописаний функціонал до певного філіалу.

4.2 Структура бази даних

Оскільки дана система навчання робототехніці є досить складною та такі дані, як конфіденційні дані користувачів, групи, напрями навчання, новини, то існує потреба у постійному збереженні інформації у базі даних.

Дана система використовує реляційну базу даних MySQL. Дана система управління базами даних є відмінним рішенням для даної системи, оскільки вона існує вже майже 15 років, що підкреслює її надійність. Також ключову роль відіграє життєвий цикл веб-орієнтованої системи, оскільки вона напрямлена на тривалий час існування.

В поточній базі даних існує 13 таблиць (рис. 4.9).

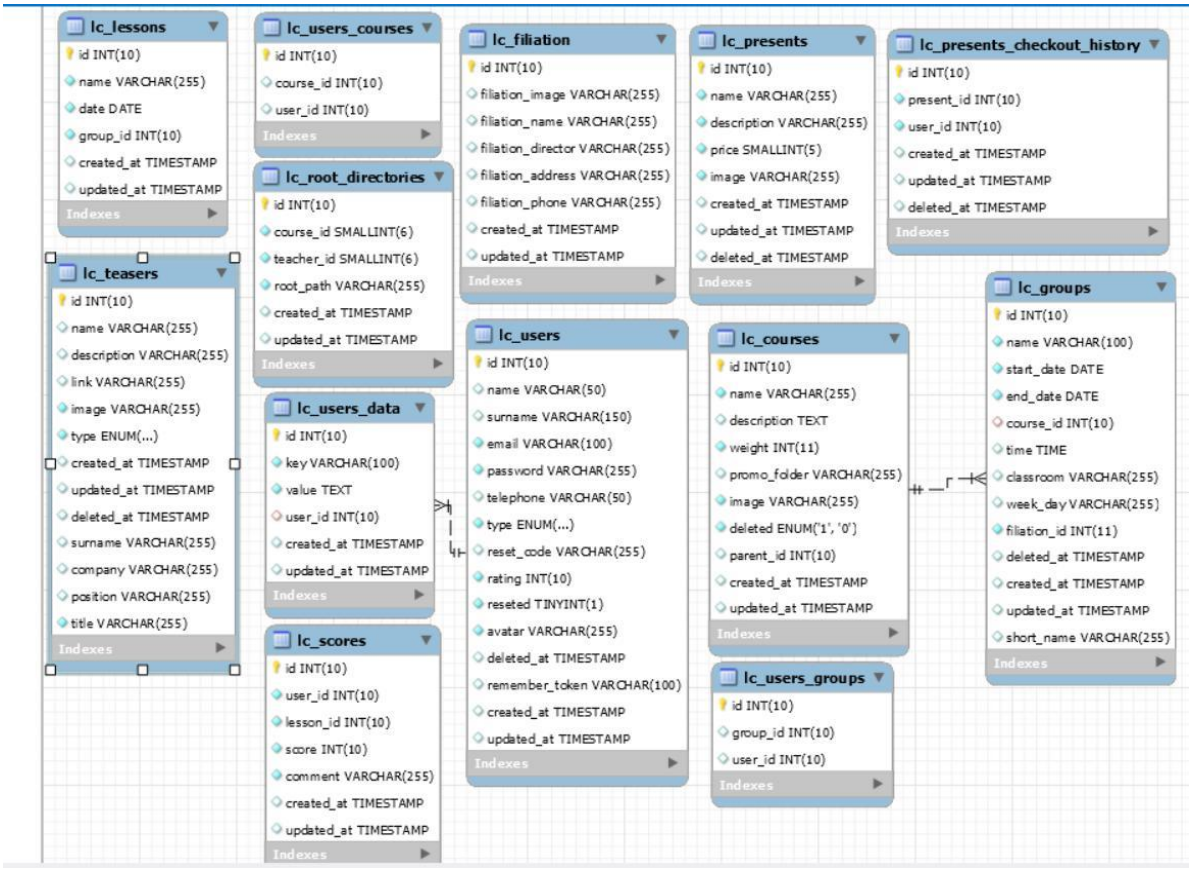


Рисунок 4.9 – ER-діаграма схеми бази даних

До складу схеми входять наступні таблиці:

1) `lc_courses` – таблиця, в якій зберігаються назви напрямів навчання та їх описи, дати створення та оновлення;

2) `lc_filiation` – таблиця, в якій зберігається інформація щодо всіх філіалів, наприклад, адреса, номеру телефону, повне ім’я директору;
3) `lc_groups` – таблиця, в якій зберігається інформація, що стосується груп, які навчаються у навчальному центрі. Основною інформацією є назва, початок та кінець періоду навчання, напрям навчання, час та день проведення занять;
4) `lc_lessons` – таблиця, у якій зберігається мета інформація щодо уроків, наприклад, назва заняття, дата створення та ідентифікатор групи, до якої відноситься;
5) `lc_presents` – таблиця, яка містить інформацію щодо подарунків, які студенти можуть отримати у якості нагороди за добре навчання;
6) `lc_presents_checkout_history` – таблиця, що містить дані стосовно подарунків, які вже були видані найкращим студентам. Містить ідентифікатори подарунку, який було обрано, та ідентифікатор користувача, якому було видано;
7) `lc_root_directories` – таблиця, що є маршрутизатором до директорії з методичними матеріалами. В ній зберігаються ідентифікатори напрямку навчання та вчителя. Це необхідно для нормального функціонування модуля формування методичних матеріалів;
8) `lc_scores` – таблиця, яка містить інформацію щодо оцінок студентів. Кожний запис містить ідентифікатори користувача та заняття, до якого прив’язана оцінка;
9) `lc_teasers` – таблиця, що містить інформацію щодо новин, заходів, акцій. До інформації належать назва, опис, тип, дати створення та оновлення;
10) `lc_users` – є найбільш важливішою таблицею, в якій зберігається вся інформація стосовно користувачів, наприклад, ім’я, фамілія, поштова адреса та пароль у хешованому виді, номер

телефону, тип, дати створення та оновлення. Більшість інших таблиць зберігають ідентифікатори записів цієї таблиці;

11) `lc_users_courses` – таблиця, яка використовується для зв’язки користувачів та їх напрямку навчання. Використовується для фільтрації задля отримання поштових адрес, методичних матеріалів тощо. Містить лише ідентифікатори користувача та напрямку навчання;
12) `lc_users_data` – у цій таблиці зберігається додаткова інформація щодо користувачів, наприклад, стать, дата народження, місто проживання та ідентифікатор користувача, до якого ця інформація відноситься;
13) `lc_users_groups` – ця таблиця містить співставлення користувача та групи, до якої він відноситься. Зберігає лише два ідентифікатори: користувача та групи.

4.3 Діагностика системи на наявність вразливостей

4.3.1 Тестування системи на SQL Injection вразливість

Для того, щоб перевірити систему на дану вразливість необхідно у формі для авторизації ввести “' OR 1=1 --” у поля для поштової адреси та паролю.

Метою даної строки є вплив на запит до бази даних, який дозволить авторизуватись першим користувачем, або якщо відома поштова скринька, то певним користувачем. Є шанс того, що даним користувачем буде адміністратор і тоді зловмисник матиме доступ до всіх даних.

Результат наведено на рис. 4.10.

Захист впроваджено на клієнтському інтерфейсі, але зловмисник може використовувати спеціальні засоби для створення запитів, наприклад Postman або той же браузер Mozilla Firefox відредагувавши попередній запит.

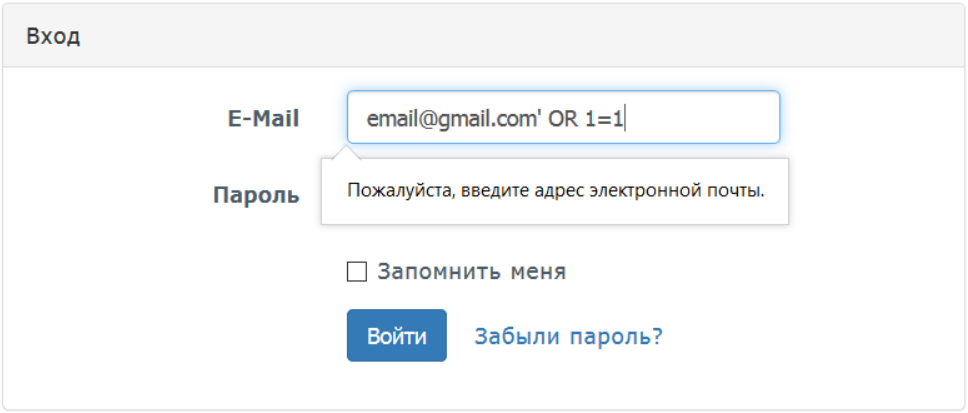


Рисунок 4.10 – Спроба впровадження SQL-скрипту у поле форми авторизації

Результат виконання запиту з використанням зміни параметрів наведено на рис. 4.11 з використанням браузеру Mozilla Firefox.

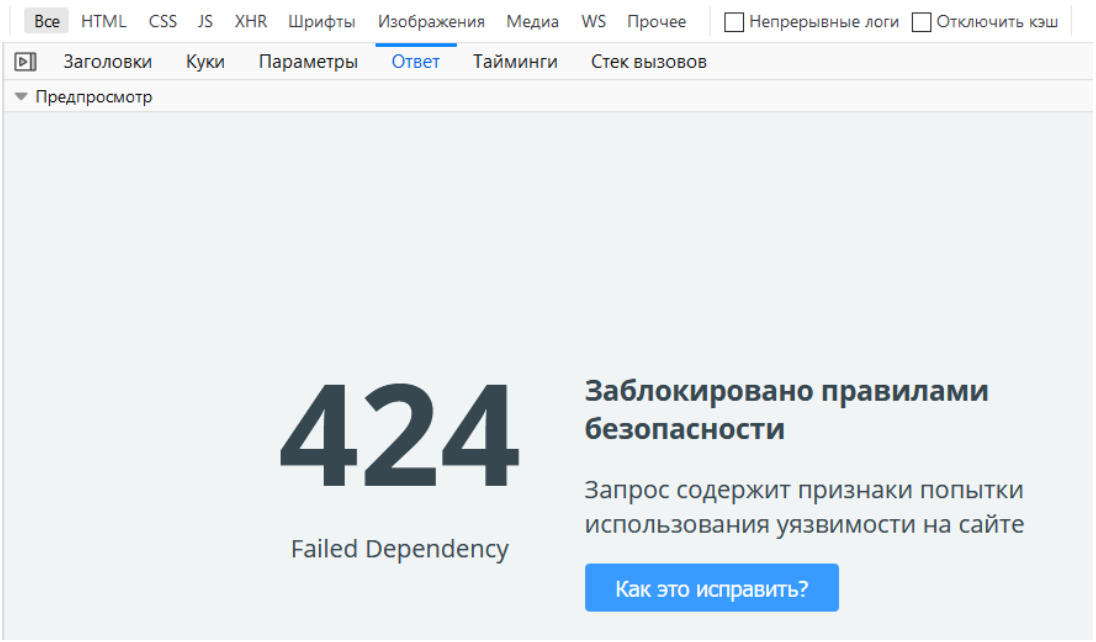


Рисунок 4.11 - Результат виконання запиту з використанням зміни параметрів

Оскільки поля для авторизації є єдиною можливістю для користувацького вводу, то можна зробити висновок, що система має достатній захист для цього виду атак.

4.3.2 Тестування системи на Brute force вразливість

Даний вид атаки передбачає перебір паролів. Його складність залежить від всієї кількості можливих варіантів. Більшість веб-орієнтованих систем мають впроваджене блокування від даного виду атаки.

В даній системі таке блокування відсутнє. Також, завдання спрощується за рахунок того, що пароль може містити лише цифри.

Для тестування на даний вид вразливості використовується програма Brutus AET2 та фейковий акаунт з тимчасовою електронною скринею. Для того, щоб почати підбор паролю необхідно знати хост та порт серверу, email або логін користувача та обрати діапазон значень.

Результат виконання зображено на рис. 4.12.

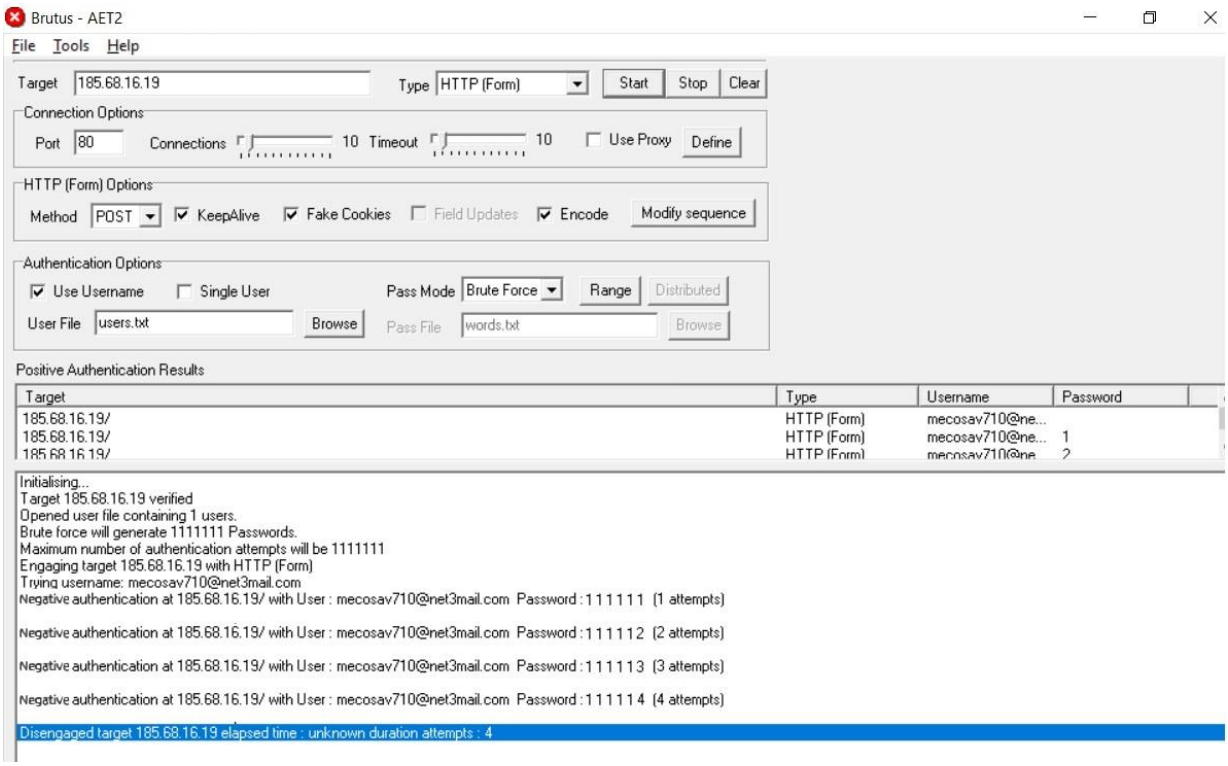


Рисунок 4.12 – Приклад проведення атаки за допомогою програми Brutus AET2

Після атаки було отримано вірний пароль, що свідчить про актуальність атаки для даної системи.

4.3.3 Тестування системи на мережеву DoS вразливість

Оскільки даний вид атаки може привести шкоду для користувачів веб-орієнтованої системи, то тестування буде проводитись на локальній машині.

Для того, щоб перевірити стійкість системи, буде використовуватись програма LOIC.

Щоб розпочати атаку, потрібно знати URL або IP-адресу веб-орієнтованої системи, порт та протокол передачі даних (рис. 4.13).

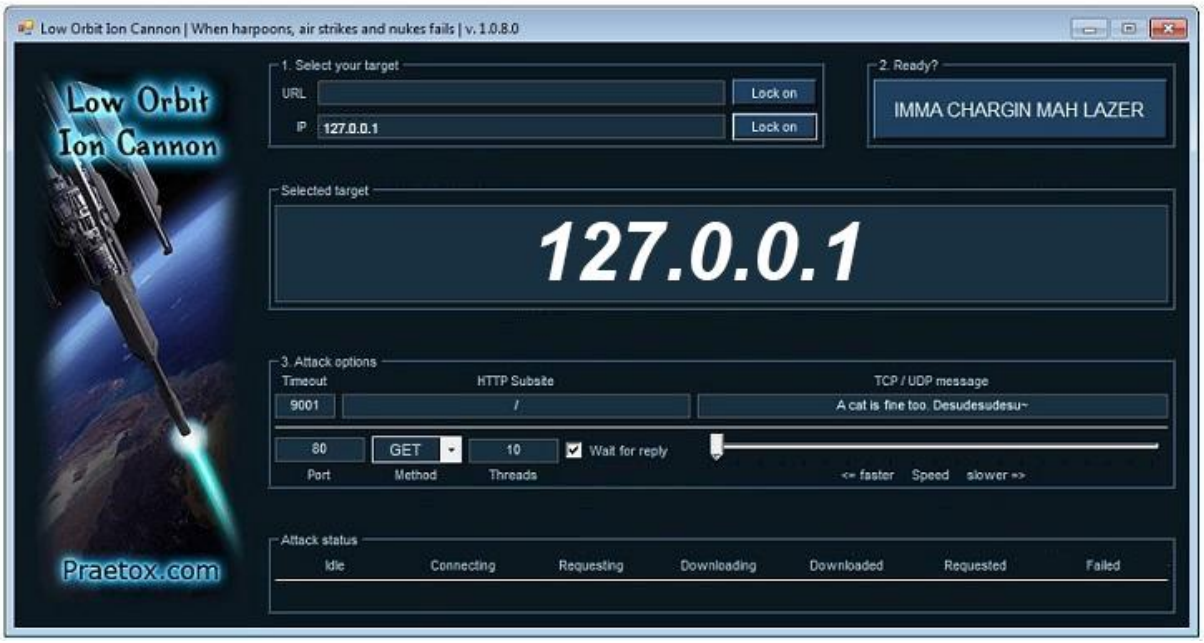


Рисунок 4.13 – Приклад конфігурації для проведення мережевої DoS-атаки на систему

Результат тестування на стійкість від мережевої DoS атаки зображено на рис. 4.14.

Таким чином, можна зробити висновок, що система не має захисту проти даного виду загрози.

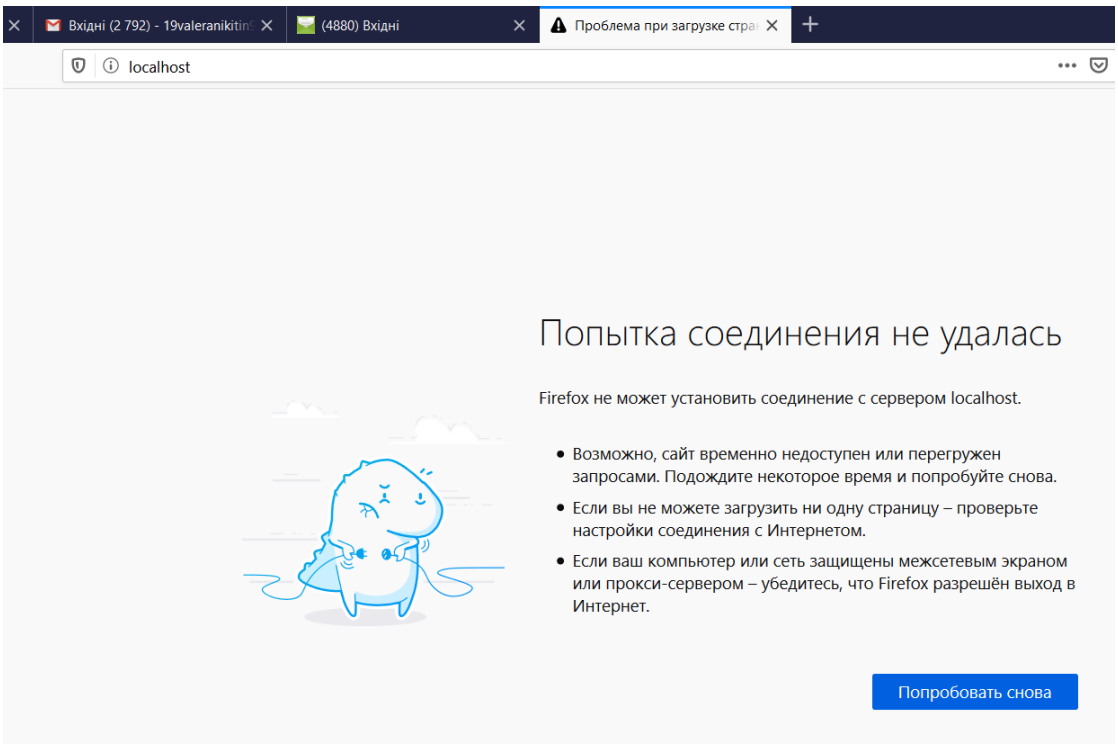


Рисунок 4.14 – Результат тестування системи на стійкість від мережевої DoS-атаки

4.3.4 Тестування системи на складність перехоплення конфіденційних даних на стороні клієнта

При заповнені форми авторизації та відправці даних на сервер можливий витік інформації, оскільки дані для авторизації відправляються на сервер у незашифрованому вигляді. Хоча в якості протоколу використовується HTTPS, це не є достанім, оскільки ці дані можуть бути перехоплені з використанням програм-сніфферів, наприклад у публічних точках роздачі WI-FI.

В даній системі не використовується шифрування на стороні клієнта (рис. 4.15), оскільки дані відправляються у незашифрованому вигляді.

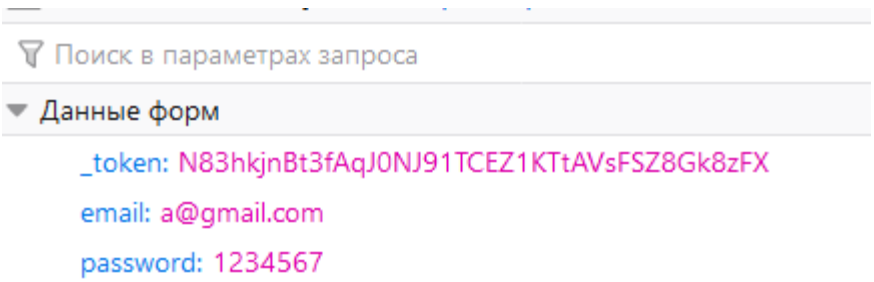


Рисунок 4.15 – Дані, що відправляються на сервер при авторизації

Слід зазначити, що для роботи із системою використовується протокол HTTPS, який забезпечує шифрування даних, що передаються. Але це не є достатнім, оскільки даний протокол має кілька суттєвих вразливостей, що дозволяють відносно легко отримати вихідні дані.

4.4 Алгоритми необхідних функцій для підвищення безпеки системи навчання робототехніці

4.4.1 Алгоритм роботи функції для попередження Brute force атаки

Для попередження даної атаки достатньо фіксувати кількість спроб авторизації з певної IP-адреси та зробити блокування на кілька хвилин, якщо кількість спроб перевищуватиме допустиме значення. Це дозволить зірвати атаку, оскільки після закінчення строку блокування зловмисник вимушений буде розпочати її заново(рис. 4.17).

4.4.2 Алгоритм роботи функції для попередження мережевої DoS-атаки

Задля підвищення захисту веб-орієнтованої системи навчання робототехніці від мережевої DoS атаки запропоновано наступний алгоритм (рис. 4.18):

1) при відправці запиту клієнтом, необхідно зі спеціально створеної таблиці схему бази даних видалити всі записи, що були створені пізніше, ніж 2 секунди від часу теперішнього запиту;
2) занести у цю ж таблицю IP-адресу та час поточного запиту;
3) дістати з таблиці чотири останніх записи, які зв’язані з поточною IP-адресою;
4) якщо кількість записів не дорівнює 4, тоді пропустити на подальшу обробку запиту;
5) якщо ж кількість дорівнює 4, то необхідно зіставити різницю між останнім запитом та першим з 1 секундою. Якщо значення різниці

менша за 1 секунду, тоді необхідно заблокувати поточну IP-адресу, додавши її у чорний список на сервері;

6) Якщо різниця більша, або дорівнює одній секунді, тоді пропустити поточний запит на подальшу обробку.

4.4.3 Алгоритм роботи функції для попередження перехоплення конфіденційних даних на стороні клієнта

Для підвищення безпеки конфіденційних даних користувача, що авторизується, можна використовувати додаткове асинхронне шифрування (рис. 4.16).

Рисунок 4.16 – Блок-схема алгоритму для зниження ризику перехоплення конфіденційних даних

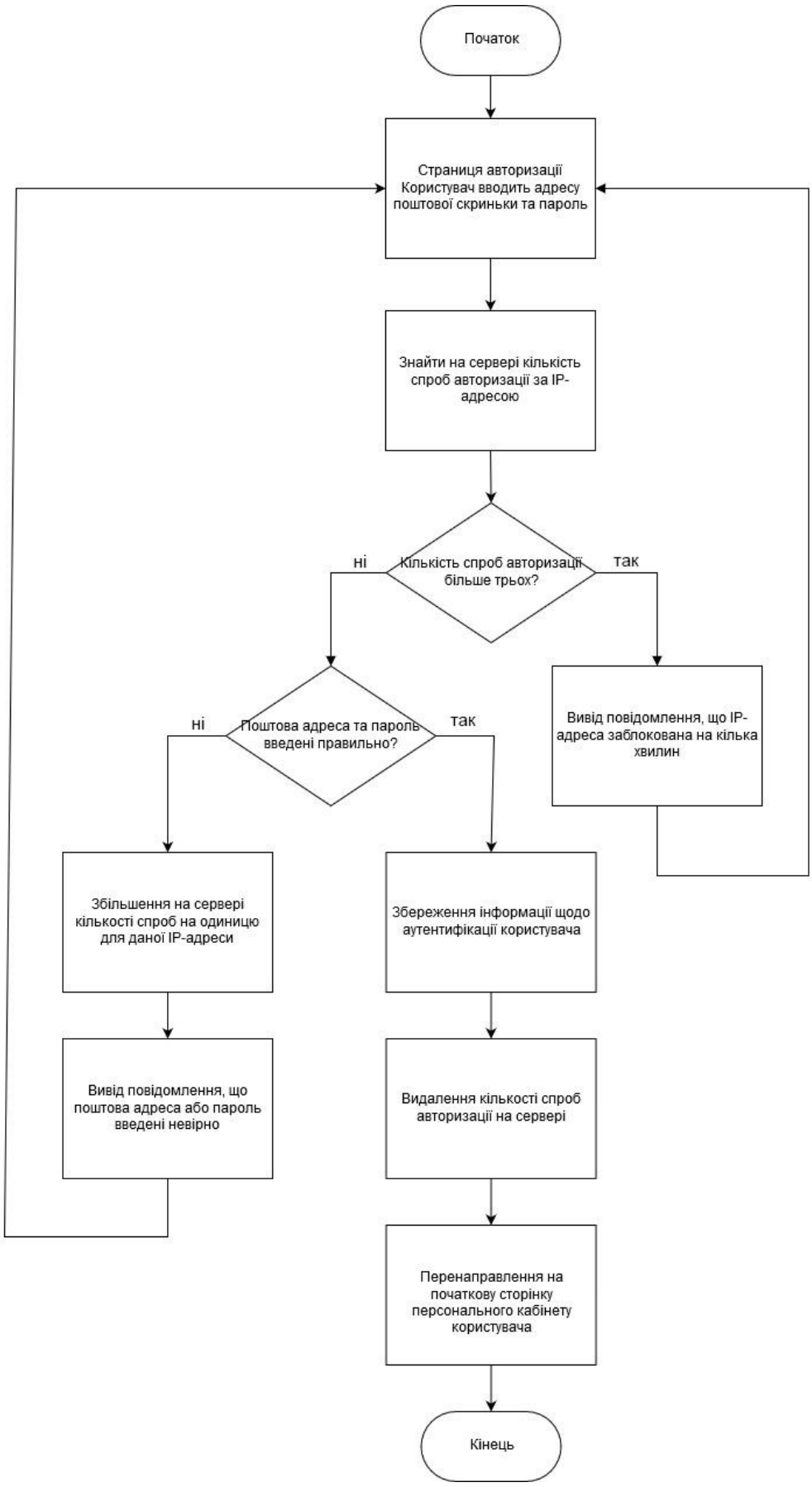


Рисунок 4.17 – Блок-схема алгоритму функції попередження для проведення мережевої DoS-атаки

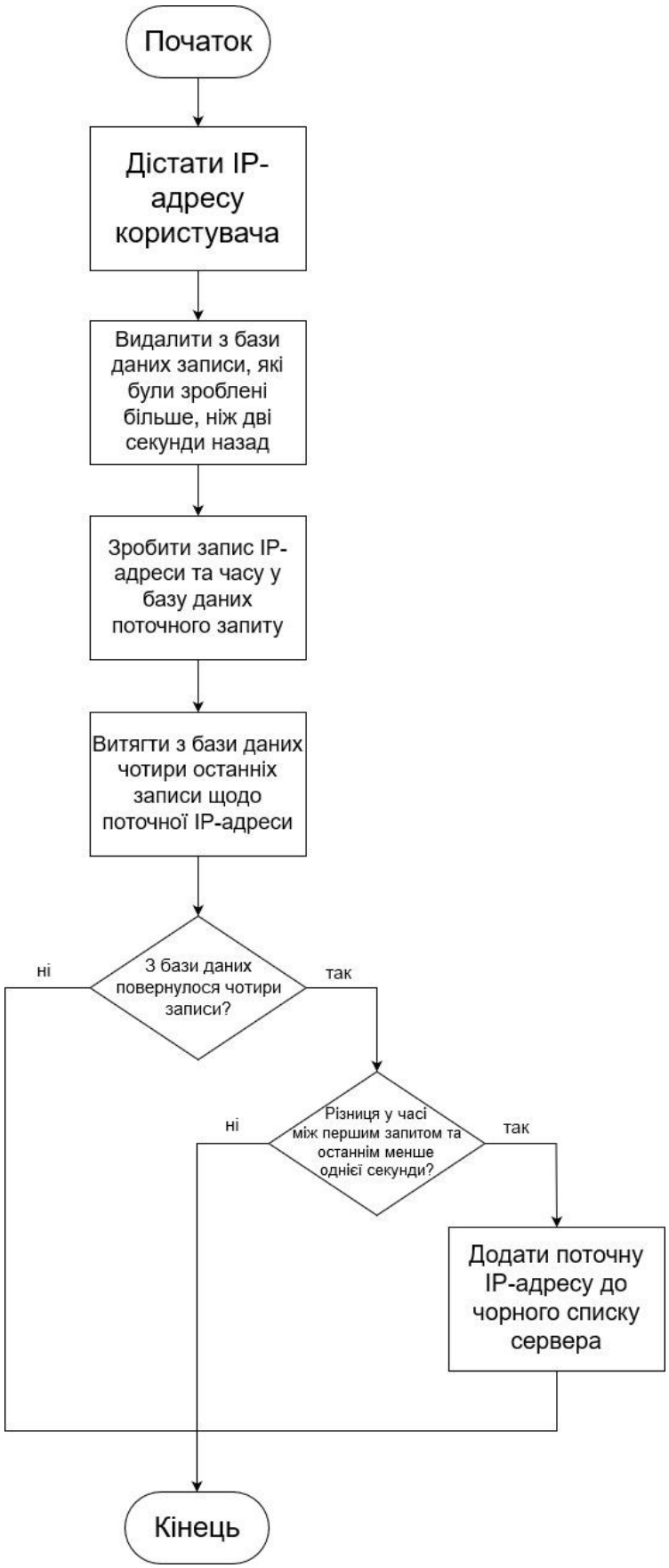


Рисунок 4.18 – Блок-схема алгоритму роботи функції для попередження мережевої DoS-атаки

4.3 Реалізація підсистеми захисту для фреймворку Laravel

4.3.1 Реалізація функції для забезпечення захисту від Brute force атаки

Використовуючи розроблений алгоритм, що зображено на рис. 4.16, було реалізовано функцію з використанням мови програмування PHP. Програмний код наведено на рис. 4.19.

```
protected function credentials(Request $request)
{
    $apc_key = "{$_SERVER['SERVER_NAME']}~login:{$_SERVER['REMOTE_ADDR']}";
    $apc_blocked_key = "{$_SERVER['SERVER_NAME']}~login-blocked:{$_SERVER['REMOTE_ADDR']}";

    $tries = (int)apc_fetch($apc_key);
    if ($tries > 3) {
        header("HTTP/1.1 429 Too Many Requests");
        $data['errors']['email'] = "You've exceeded the number of login attempts.
        We've blocked IP address {$_SERVER['REMOTE_ADDR']} for a few minutes.";
        return $data;
    }

    $success = login($_POST['username'], $_POST['password']);
    if (!$success) {
        $blocked = (int)apc_fetch($apc_blocked_key);
        apc_store($apc_key, $tries+1, pow(2, $blocked+1)*60);
        apc_store($apc_blocked_key, $blocked+1, 86400);
    } else {
        apc_delete($apc_key);
        apc_delete($apc_blocked_key);
    }
    $data = $request->only($this->username(), 'password');
    $data['reseted'] = false;
    return $data;
}
```

Рисунок 4.19 – Впроваджений код для запобігання проведення Brute force атаки

Результат тестування після впровадження підсистеми зображено на рис. 4.20 та 4.21 відповідно.

Через дві хвилини блокування знімається (рис. 4.22).

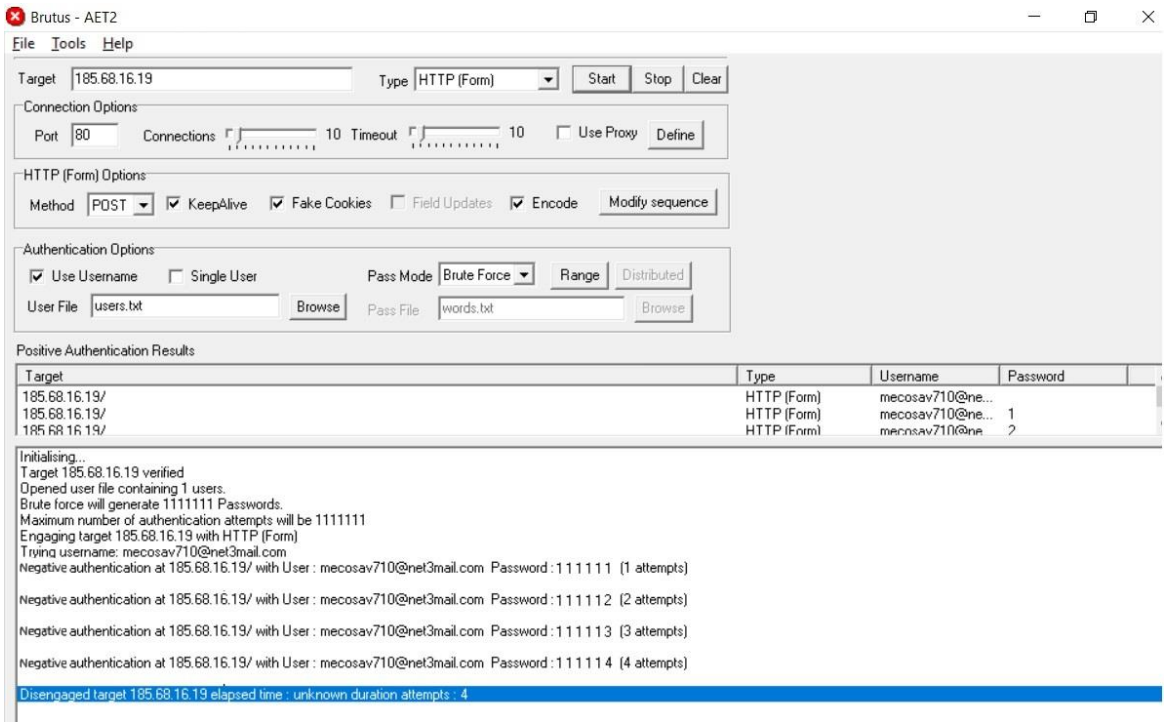


Рисунок 4.20 – Спроба підбору пароля до акаунта при реалізованій підсистемі захисту від Brute force атаки

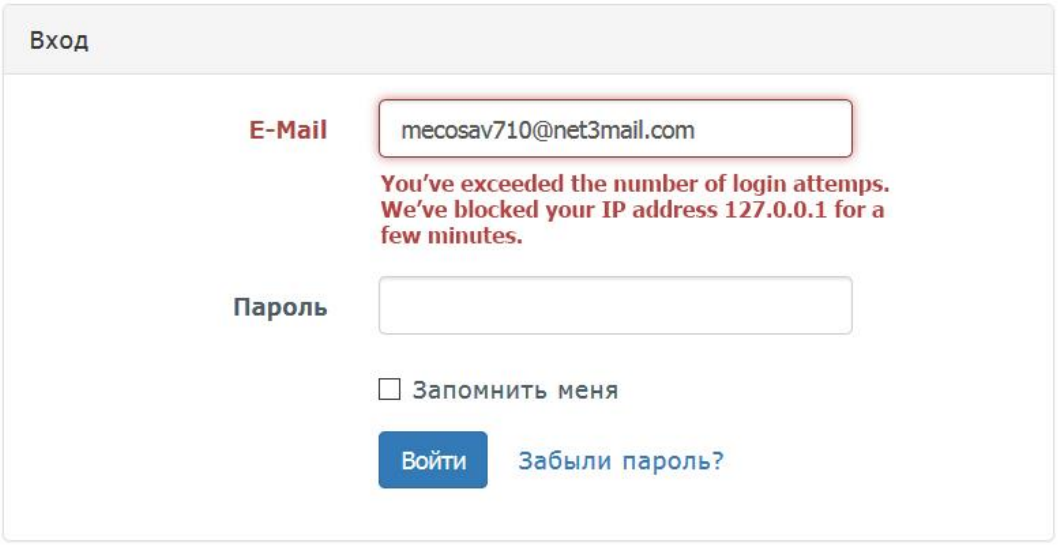


Рисунок 4.21 – Відображення повідомлення на стороні клієнта при блокуванні IP-адреси

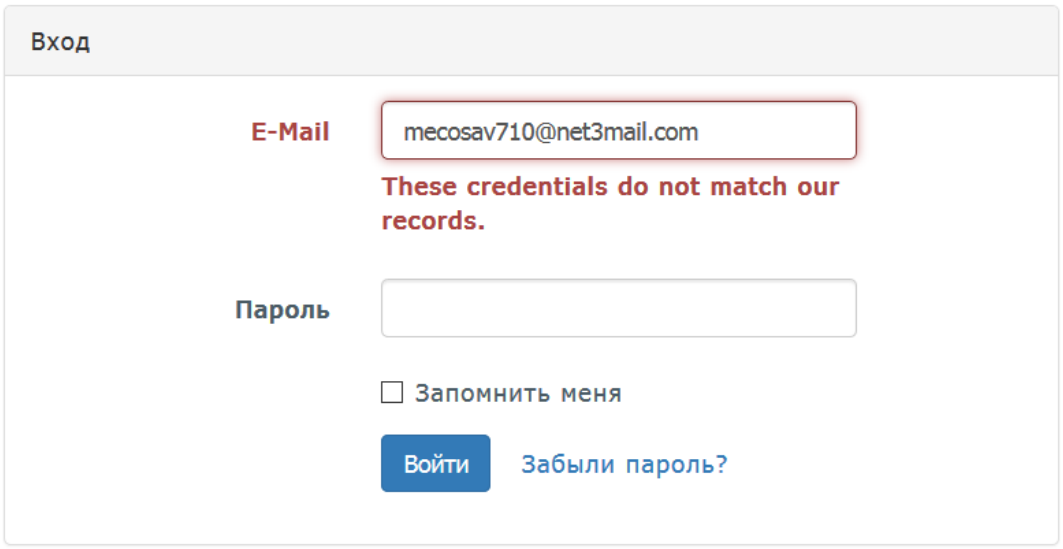


Рисунок 4.22 – Результат спроби авторизації після закінчення терміну блокування

4.3.2 Реалізація функції для забезпечення захисту від мережевої DoS-атаки

Виправити вразливість можна блокуванням IP-адрес, у котрих частота запитів набагато більше, ніж зазвичай потрібно для роботи з системою. Також необхідно створити у базі даних додаткову таблицю, яка буде містити тимчасові дані про користувачів.

Таблиця створюється SQL-скриптом, який зображено на рис. 4.23.

```
62 • CREATE TABLE `access_log` (
63       `ip` varchar(15) NOT NULL default '',
64       `enter_time` int(11) NOT NULL default '0'
65   ) ENGINE=HEAP;
```

Рисунок 4.23 – SQL-скрипт створення таблиці для збереження інформації щодо IP-адреси та часу активності користувача

Реалізація функції для запобігання мережевої DoS-атаки зображена на рис. 4.24.

```php
function check_ddos(){
    $rec_limit = 4;
    $time_limit = 1;
    $ip = $_SERVER['REMOTE_ADDR'];

    if (!$db_connection = mysql_pconnect('localhost', 'root', '')){
        die();
    }
    mysql_query('DELETE FROM access_log WHERE enter_time < '.(time() - ($time_limit * 2)), $db_connection);
    mysql_query('INSERT INTO access_log (ip, enter_time) VALUES ("'.$ip.'", '.time().')', $db_connection);
    if ($result = mysql_query('SELECT enter_time FROM access_log WHERE ip="'.$ip.'"
    ORDER BY enter_time DESC LIMIT '.$rec_limit)){
        while($row = mysql_fetch_row($result)){
            $result_array[] = $row;
        }
    }
    $rec_count = count($result_array);
    if ($rec_count == $rec_limit){
        $first_time = $result_array[$rec_count - 1][0];
        $last_time = $result_array[0][0];
        if (($last_time - $first_time) < $time_limit){
            iptables_ban_ip($ip);
            exit();
        }
    }
}
```

Рисунок 4.24 – Реалізація функції для захисту від мережевої DoS-атаки

Під час другої спроби зробити мережеву DoS-атаку, система блокує IP-адресу та не дає доступу користувачу.

Результат впровадження зображено на рис. 4.25.

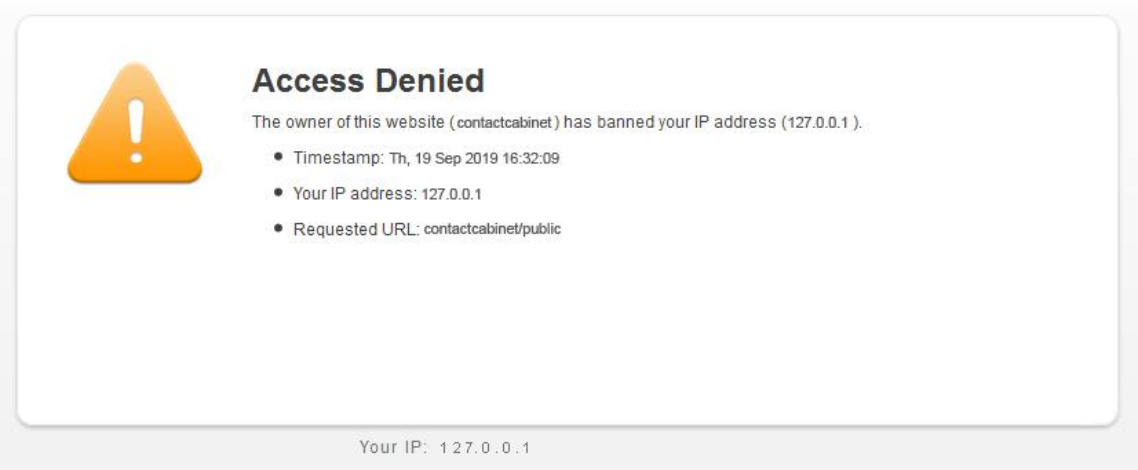


Рисунок 4.25 – Результат роботи впровадженої підсистеми від мережевої DoS-атаки

4.3.3 Реалізація функції для підвищення захисту конфіденційних даних користувача при авторизації

Для підвищення безпеки буде використано алгоритм, що зображено на рис. 4.18 та бібліотека crypto-js. В якості алгоритму шифрування буде

використано адаптивну криптографічну хеш-функцію bcrypt, оскільки на сервері використовується саме цей алгоритм шифрування. Це дає змогу легко інтегрувати існуюче рішення з впровадженим кодом (рис. 4.26).

```
<script>
    $(function () {
        $("#pass").on("input",function(e){
            var rounds = 10;
            var bcrypt_pass=bcrypt.hash($('#pass').val(),rounds, false);
            var bcrypt_email=bcrypt.hash($('#email').val(),rounds, false);
            $('#hidpass').val(bcrypt_pass.toString());
            $('#hidemail').val(bcrypt_email.toString());
        });
    });
</script>
```

Рисунок 4.26 – Підсистема для шифрування даних на стороні клієнта

Хеш, який буде отримано цією функцію можна напряму використовувати для порівняння с хешем, що зберігається в базі даних.

Результат шифрування зображено на рис. 4.27.

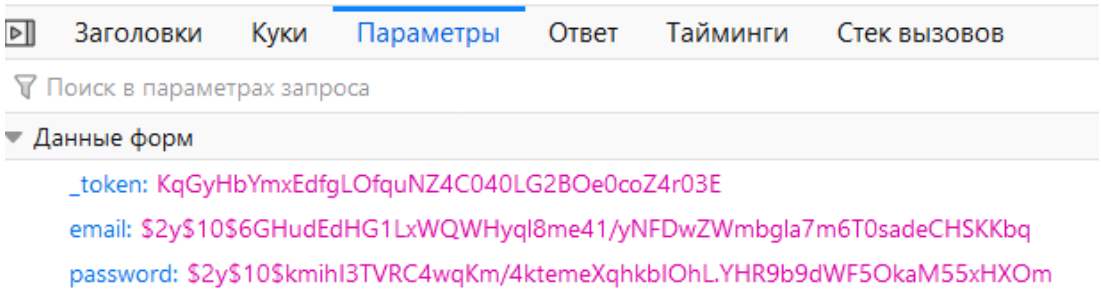


Рисунок 4.27 – Результат шифрування даних на стороні клієнта

4.3.4 Створення векторів оптимізуючих перетворень та вибір оптимального для системи навчання робототехніці

За умовами технічного завдання, основна вимога припадає на час відгуку. Цей показник не має зрости більше ніж на 30%.

Для того, щоб перевірити це, необхідно зробити запити до всіх можливих сторінок системи. Вимірювання часу буде відбуватися на локальній машині, оскільки це дасть можливість оцінити саме серверну частину веб-орієнтованої системи без урахування затрат часу на передавання та отримання інформації по мережі. Для цього буде використана програма Postman (рис. 4.28).

Результати вимірювань наведені у табл. 4.1.

Таблиця 4.1 – Час відгуку системи на запити

Назва сторінки	Час відгуку (без впроваджених змін), ms	Час відгуку (з впровадженим кодом для захисту від Brute force), ms	Час відгуку (з впровадженим кодом для захисту від мережевої DoS-атаки), ms	Час відгуку (з впровадженим кодом для захисту від перехоплення мережевого трафіку), ms
Головна	339	342	435	334
Про нас	60	63	164	59
Наша команда	30	28	129	33
Ціни	32	68	98	35
Дитяча IT Академія	43	41	142	44
Контакти	34	35	135	34
Розклад	68	62	167	65
Авторизація	128	171	126	160

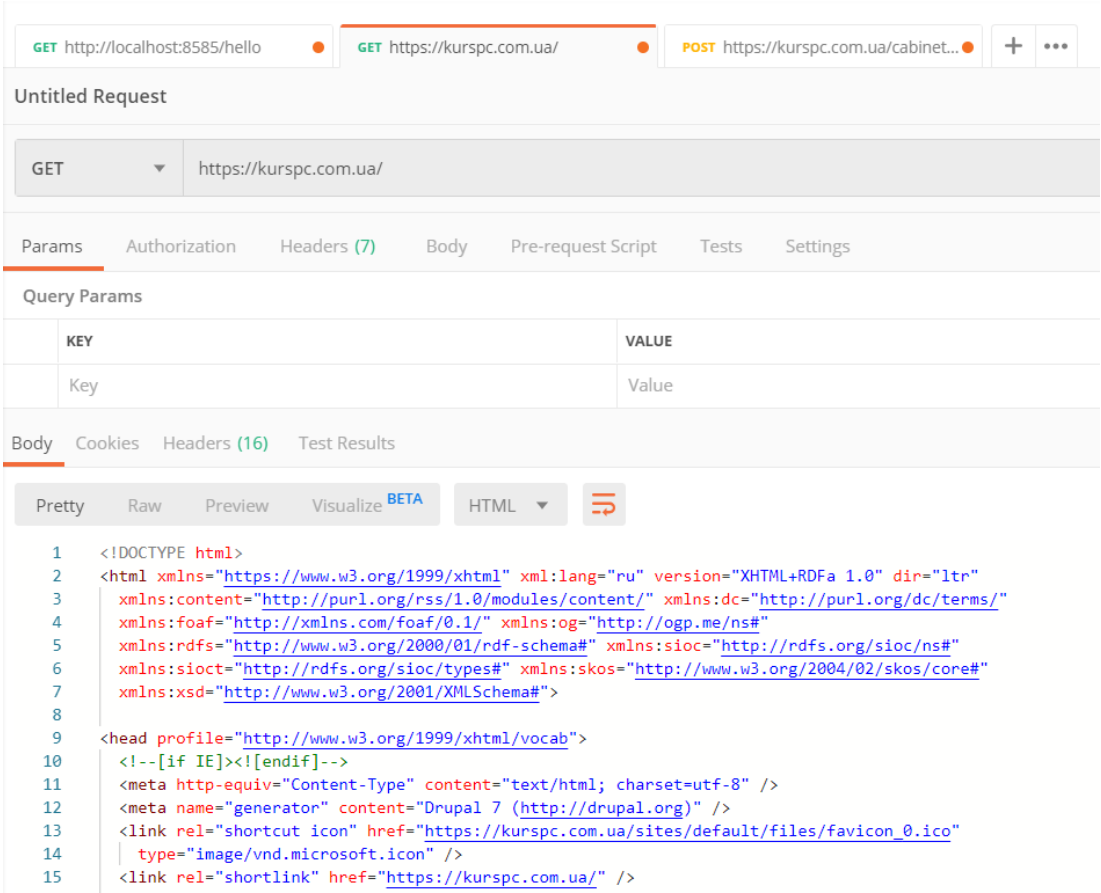


Рисунок 4.28 – Програма для створення запитів

Оптимізаційні перетворення можна комбінувати між собою, тим самим отримуючи вектори оптимізуючих перетворень (табл. 4.2).

Таблиця 4.2 – Вектори оптимізуючих перетворень

Оптимізуюче перетворення / Вектор оптимізуючих перетворень	O_1	O_2	O_3
V_0	0	0	0
V_1	1	1	1
V_2	1	0	0
V_3	0	1	0
V_4	0	0	1
V_5	1	1	0
V_6	0	1	1
V_7	1	0	1

Оскільки мається обмеження за часом, то необхідно розрахувати на скільки кожен вектор оптимізуючих перетворень зменшує швидкість роботи системи.

Використовуючи дані табл. 4.1, розрахуємо сумарний час для кожного вектору оптимізуючих перетворень, необхідний серверу для генерації відповіді за формулою 3.4:

$$T(V_0) = 339 + 60 + 30 + 32 + 43 + 34 + 68 + 128 = 734 \text{ мс};$$

$$T(V_1) = 435+164+129+98+142+135+167+192 = 1462 \text{ мс};$$

$$T(V_2) = 342+63+28+68+41+35+62+171 = 810 \text{ мс};$$

$$T(V_3) = 435+164+129+98+142+135+167+126 = 1396 \text{ мс};$$

$$T(V_4) = 334+59+33+35+44+34+65+160 = 764 \text{ мс};$$

$$T(V_5) = 435+164+129+98+142+135+167+171 = 1439 \text{ мс};$$

$$T(V_6) = 435+164+129+98+142+135+167+160 = 1430 \text{ мс};$$

$$T(V_7) = 339+60+30+62+43+34+68+192 = 828 \text{ мс}.$$

Отриманий час $T(V_0)$ будемо вважати еталоном, оскільки це час, який необхідний серверу для генерації відгуку без оптимізуючих перетворень.

Тепер необхідно розрахувати приріст у часі для генерації сторінок сервером системи в залежності від векторів оптимізуючих перетворень (формула 3.5).

Для розрахунку на скільки вектори оптимізуючих перетворень вплинуть на швидкодію системи використаємо формулу 3.5:

$$\Delta T(V_1) = \frac{(1462 - 734)}{734} \cdot 100 = 99.18\ \%;$$

$$\Delta T(V_2) = \frac{(810 - 734)}{734} \cdot 100 = 10.35\ \%;$$

$$\Delta T(V_3) = \frac{(1396 - 734)}{734} \cdot 100 = 90.19\ \%;$$

$$\Delta T(V_4) = \frac{(764 - 734)}{734} \cdot 100 = 4.08\ \%;$$

$$\Delta T(V_5) = \frac{(1439 - 734)}{734} \cdot 100 = 96.04\ \%;$$

$$\Delta T(V_6) = \frac{(1430 - 734)}{734} \cdot 100 = 94.82\ \%;$$

$$\Delta T(V_7) = \frac{(828 - 734)}{734} \cdot 100 = 12.8\ \%.$$

Використовуючи формулу 3.6, значення табл. 4.2 та розраховані значення впливу на швидкодію системи для кожного вектору, розрахуємо коефіцієнт захисту для кожного вектору оптимізуючих перетворень:

$$P_1 = 0;$$

$$P_2 = 0.52 \cdot 1 + 0.74 \cdot 0 + 0.63 \cdot 0 = 0.52;$$

$$P_3 = 0;$$

$$P_4 = 0.52 \cdot 0 + 0.74 \cdot 0 + 0.63 \cdot 1 = 0.63;$$

$$P_5 = 0;$$

$$P_6 = 0;$$

$$P_7 = 0.52 \cdot 1 + 0.74 \cdot 0 + 0.63 \cdot 1 = 1.15.$$

Отримані результати зведені у табл. 4.3.

Таблиця 4.3 – Результати розрахунків коефіцієнтів захисту та погіршення швидкодії системи внаслідок впроваджених функцій

Вектор	Коефіцієнт захисту	Зменшення швидкодії системи, %
V_1	0	99.18
V_2	0.52	10.35
V_3	0	90.19
V_4	0.63	4.08
V_5	0	96.04
V_6	0	94.82
V_7	1.15	12.8

За формулою 3.7, найоптимальнішим вектором оптимізуючих перетворень серед отриманої множини значень коефіцієнту захисту є вектор V_7.

Висновки за розділом

У даному розділі описано схему реалізованої системи навчання робототехніці та її структуру бази даних. Також були представлені блок-схеми алгоритмів, які були запропоновані, задля нейтралізації існуючих вразливостей фреймворку Laravel, які були знайдені під час діагностики існуючої системи навчання робототехніці. Знайдені вразливості були знешкоджені шляхом впровадження функцій на мові програмування PHP та використання бібліотеки crypto.js. З використанням запропонованого способу розрахунку захищеності системи, обрано оптимальний вектор оптимізуючих перетворень, який забезпечує найбільший захист та задовольняє умові у погіршенні швидкодії системи загалом.

РОЗДІЛ 5 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

5.1 Опис ідеї проекту

Для виконання даного підпункту було розглянуто та проаналізовано, а також зведено у таблицях:

1) вміст ідеї;
2) можливі напрямки застосування;
3) основні вигоди, що може отримати користувач товару (за кожним напрямом застосування);
4) основні відмінності від вже існуючих рішень.

Аналіз потенційних техніко-економічних переваг ідеї (чим відрізняється від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів передбачає:

— визначення переліку техніко-економічних властивостей та характеристик ідеї;

— визначення попереднього кола конкурентів (проектів-конкурентів) або товарів-замінників чи товарів-аналогів, що вже існують на ринку, та проводиться збір інформації щодо значень техніко-економічних показників для ідеї власного проекту та проектів-конкурентів відповідно до визначеного вище переліку;

— проводиться порівняльний аналіз показників: для власної ідеї визначаються показники, що мають гірші значення (W, слабкі), аналогічні (N, нейтральні) значення, кращі значення (S, сильні) (табл. 5.2).

Визначений перелік слабких, сильних та нейтральних характеристик та властивостей ідеї потенційного товару є підґрунтям для формування його конкурентоспроможності[16].

Таблиця 5.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розроблену підсистему можна застосувати до багатьох веб-орієнтованих систем, що були створені за допомогою фреймворку Laravel для підвищення безпеки	1. Фінансові організації, що надають можливість інтернет-банкінгу	Зменшення вірогідності відмови у обслуговуванні
	2. Інтернет-магазини	Зменшення вірогідності витоку конфіденційних даних
	3. Малий та середній бізнес, що мають веб-орієнтовані системи з власною базою даних	Зменшує ризик злому акаунтів

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

Техніко-економічні характеристики ідеї	Товари/концепції конкурентів		W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
	Мій проект	HTML Purifier			
Простота використання	Використання зводиться до виклику необхідних функцій у потрібному місці	Необхідність використовувати у якості віджету	+		
Рівень захисту	Забезпечує безпеку системи від найбільш розповсюджених атак	Забезпечує безпеку системи від кількох видів атак			+
Залежність від розміру проекту	Не сприяє ускладненню при розширенні функціоналу	Уповільнює роботу системи при багатократному використанні			+
Адаптивність до інших фреймворків	Може адаптуватись	Може адаптуватись		+	
Розмір	Невеликий розмір вихідного коду	Суттєвий розмір вихідного коду			+

5.2 Технологічний аудит ідеї проекту

В межах даного підрозділу проведено аудит технології, за допомогою якої можна реалізувати ідею проекту (технології створення товару). Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових (табл. 5.3):

а) за якою технологією буде виготовлено товар згідно ідеї проекту;

б) чи існують такі технології, чиїх потрібно розробити/доробити;

в) чи доступні такі технології авторам проекту.

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
Розроблення підсистеми для підвищення безпеки веб-орієнтованих систем, що були створені за допомогою фреймворку Laravel	Фреймворк Laravel	Наявна	Доступна
	Алгоритми захисту від мережевих атак	Наявна	Доступна
	Мова програмування PHP	Наявна	Доступна

За результатами аналізу таблиці зроблено висновок щодо можливості технологічної реалізації проекту. Технологічним шляхом реалізації проекту було обрано такі технології, як фреймворк Laravel та мова програмування PHP через їх доступність та безкоштовність.

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів[17]. Спочатку було проведено аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (табл. 5.4).

Середню норму рентабельності в галузі було порівняно із банківським відсотком на вкладення. Останній є меншим, тому є сенс вкладати гроші саме у цей проект.

За результатами аналізу табл. 5.4 зроблено висновок, що ринок є привабливим для входження.

Надалі були визначені потенційні групи клієнтів, їх характеристики та сформовано орієнтовний перелік вимог до товару для кожної групи (табл. 5.5).

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап-проекту

Показники стану ринку (найменування)	Характеристика
Кількість головних гравців, од	1
Загальний обсяг продаж, грн/ум. од	25000
Динаміка ринку (якісна оцінка)	Зростає
Наявність обмежень для входу (вказати характер обмежень)	-
Специфічні вимоги до стандартизації та сертифікації	-
Середня норма рентабельності у галузі (або по ринку), %	18

Таблиця 5.5 – Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
Підсистема для підвищення безпеки веб-орієнтованих систем	Компанії, приватні підприємства специфіка роботи яких пов’язана з використанням Web-технологій	Відмінності у сферах діяльності компаній та приватних підприємств	Швидкість та надійність у використанні

Після визначення потенційних груп клієнтів проведено аналіз ринкового середовища: складено таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (табл. 5.6, 5.7).

Визначено можливу реакцію компанії на конкуренцію, зміну потреб користувачів та появу нових методів з врахуванням можливостей ринку та вітчизняних особливостей.

За результатами аналізу табл. 5.9 зроблено висновок про можливість роботи на ринку з огляду на конкурентну ситуацію. Також зроблено висновок щодо характеристик, які повинен мати проект, щоб бути конкурентоспроможним на ринку.

Таблиця 5.6 – Фактори загроз

Фактор	Зміст загрози	Можлива реакції компанії
Конкуренція	Вихід на ринок продуктів з кращими показниками	Вдосконалення використовуваних алгоритмів Популяризація продукту за рахунок рекламної кампанії
Зміна потреб користувачів	Користувачам необхідна підсистема, що буде забезпечувати захист від нових видів загроз	Розширення функціоналу продукту Популяризація продукту за рахунок рекламної кампанії

Таблиця 5.7 – Фактори можливостей

Фактор	Зміст можливості	Можлива реакції компанії
Конкуренція	Майже повна відсутність аналогів	Проведення рекламних заходів
Поява нових способів	Нові методи для підвищення безпеки веб-орієнтованих систем	Оптимізація існуючого рішення впровадженням нових алгоритмів та способів
Поява нових видів загроз	Нові способи для злому веб-орієнтованих систем	Розробка нових алгоритмів та їх впровадження у продукт

Надалі було проведено аналіз пропозиції: визначили загальні риси конкуренції на ринку (табл. 5.8).

Проведено аналіз конкуренції у галузі за моделлю М. Портера (табл. 5.9).

Цей висновок був врахований при формулюванні переліку факторів конкурентоспроможності у наступному пункті. На основі аналізу конкуренції, проведеного в табл. 5.9, а також із урахуванням характеристик ідеї проекту (табл. 5.2), вимог споживачів до товару (табл. 5.5) та факторів маркетингового

середовища (табл. 5.6, 5.7) визначається та обґрунтовується перелік факторів конкурентоспроможності.

Аналіз оформлено у табл. 5.10.

Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємництва (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції – монополія	На ринку присутні декілька компаній-конкурентів, але їх товар дещо відрізняється між собою.	Підтримка якості продукту та постійні вдосконалення
2. За рівнем конкурентної боротьби – міжнародний	Компанії-конкуренти з інших країн	Розробити універсальну архітектуру продукту, що дозволить легко адаптувати продукт до інших галузей
3. За галузевою ознакою – міжгалузева	Продукт може біти використаний для інших галузей	Постійне вдосконалення та оновлення продукту, що сфокусовано на первісну ціль
4. Конкуренція за видами товарів – товарно-видова	Конкуренція між видами програмного продукту, їх якістю та оновленням	Створити програмний продукт, враховуючи недоліки конкурентів та актуальність функціоналу
5. За характером конкурентних переваг – нецінова	Вдосконалення технології створення продукту для низької собівартості	Покращення моделі функціонування продукту. Використання дешевих актуальних технологій, що дозволяють дотриматись вимог якості продукту
6. За інтенсивністю – не марочна	Бренд присутній, але його роль незначна	Участь у конференціях, проведення рекламних кампаній

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких(Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (табл. 5.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (табл. 5.11). Перелік ринкових загроз та ринкових можливостей було складено на основі аналізу факторів загроз та факторів можливостей маркетингового середовища.

Таблиця 5.9 – Аналіз конкуренції у галузі за М. Портером

Складові аналізу	Прямі конкуренти у галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Навести перелік прямих конкурентів	Визначити бар'єри входження в ринок	Визначити фактори сили постачальників	Визначити фактори сили споживачів	Фактори загроз з боку замінників
	HTML Purifier	Наявність вже існуючих рішень	-	Контроль якості продукту	Авторитет конкурентів на ринку, більш широкий функціонал
Висновки	Досить інтенсивна конкурента боротьба з іншими на ринку гравцями	Є можливість виходу на ринок, але є конкуренти. Строки – 12 місяців	-	Споживачі диктують умови на ринку: надійний, швидкий та точний програмний продукт для захисту веб-орієнтованих систем	Необхідно випускати програмний продукт не гірше, ніж у конкурентів та розширяти функціонал

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для конкурентних проектів значущим)
Ціна	Більш доступна ціна збільшує кількість потенційних споживачів
Міжгалузеве використання	Дозволяє впроваджувати підсистему захисту у веб-орієнтовану систему не залежно від галузі
Актуальність	Реалізовані алгоритми захисту від найбільш актуальних загроз сучасності

За визначеними факторами конкурентоспроможності (табл. 5.10) проведено аналіз сильних та слабких сторін стартап-проекту (табл. 5.11).

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін проекту

Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні						
		-3	-2	-1	0	1	2	3
Ціна	15				+			
Міжгалузеве використання	20						+	
Актуальність	15		+					

Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення. Наприклад: зниження доходів потенційних споживачів – фактор загрози, на основі якого можна зробити прогноз щодо посилення значущості цінового фактору при виборі товару та відповідно, –цінової конкуренції (а це вже –ринкова загроза).

Таблиця 5.12 – SWOT-аналіз стартап-проекту

Сильні сторони:	Слабкі сторони:
Ціна	Авторитет на ринку
Актуальність	
Міжгалузеве використання	
Можливості:	Загрози:
Конкуренція	Зміна потреб користувача
Поява нових способів для захисту від вразливостей	

На основі SWOT-аналізу розроблено альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (див. табл. 5.9, аналіз потенційних конкурентів). Визначені альтернативи були проаналізовані з точки зору строків та ймовірності отримання ресурсів (табл. 5.13).

Таблиця 5.13 – Альтернативи ринкового впровадження стартап-проекту

Альтернатива (орієнтований комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
Безкоштовне розповсюдження створеного продукту	70%	12 місяців
Створення продукту з подальшим розповсюдженням за певну оплату	75%	12 місяців
Створення веб-орієнтованої системи, в якій можна буде користуватися продуктом	50%	18 місяців

Після аналізу було обрано альтернативу №2.

5.4 Аналіз ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: проведено опис цільових груп потенційних споживачів (табл. 5.14).

Таблиця 5.14 – Вибір цільових груп потенційних споживачів

Опис профілю цільової групи потенційних клієнтів	Готовність споживачів прийняти продукт	Орієнтований попит в межах цільової групи (сегменти)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
Компанії діяльність яких пов’язана з веб-орієнтованими системами	Висока	Високий	Сильна	Складно
Приватні підприємства міського та міжнародного рівня, діяльність яких пов’язана з веб-орієнтованими системами	Висока	Високий	Сильна	Складно
Приватні підприємства обласного рівня	Висока	Високий	Помірна	Середня складність
Підприємства регіонального характеру	Висока	Середній	Слабка	Просто

За результатами аналізу потенційних груп споживачів обрано цільової групи, для яких буде запропоновано даний товар, та визначено стратегію охоплення ринку - стратегію диференційованого маркетингу (компанія працює з декількома сегментами). Як цільові групи обрано: 1,2,3.

Для роботи в обраних сегментах ринку сформовано базову стратегію розвитку (табл. 5.15).

Таблиця 5.15 – Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Створення продукту з подальшим розповсюдженням за певну оплату	Визначити потреби кожної з груп, розробити відповідно до них стратегії приваблення клієнтів та маркетингової комунікації	Цінова політика, універсальність продукту (миттєве практичне застосування), якість та актуальність	Стратегія диференціації

Наступним кроком обрано стратегію конкурентної поведінки (табл. 5.16).

Таблиця 5.16 – Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
«першопроходець»	Забирати існуючих	Ні	Стратегія зайняття конкурентної ніші

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до продукту (див. табл. 5.5), а також в залежності від обраної базової стратегії розвитку (табл. 5.15) та стратегії конкурентної поведінки (табл. 5.16) розроблено стратегію позиціонування (табл. 5.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Таблиця 5.17 – Визначення стратегії позиціонування

Вимоги до товару цільовою аудиторією	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
Просте використання, надійність, швидкість, наявність оновлень та наявність документації для програмного продукту	Стратегія диференціації	Позиція на основі порівняння фірми з товарами конкурентів; Відмінні особливості споживача	Надійність Швидкість Просте використання

Результатом виконання підрозділу стала узгоджена система рішень щодо ринкової поведінки стартап-компанії, яка визначає напрями роботи стартап-компанії на ринку.

5.5 Розроблення маркетингової програми стартап-проекту

Сформовано маркетингову концепцію товару, який отримає споживач. Для цього у табл. 5.18 підсумовано результати попереднього аналізу конкурентоспроможності товару. Концепція товару -письмовий опис фізичних та інших характеристик товару, які сприймаються споживачем, і набору вигод, які він обіцяє певній групі споживачів.

Розроблено трирівневу маркетингову модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (табл. 5.19). 1-й рівень: при формуванні задуму товару вирішується питання щодо того, засобом вирішення якої потреби або проблеми буде даний товар, яка його основна вигода. Дане питання безпосередньо пов'язаний з формуванням технічного завдання в процесі розробки конструкторської документації на виріб. 2-й рівень: цей рівень являє рішення того, як буде реалізований товар в реальності, включає в себе якість, властивості, дизайн, упаковку, ціну. 3-й рівень: товар з підкріпленням (супроводом) – додаткові

послуги та переваги для споживача, що створюються на основі товару за задумом і товару в реальному виконанні (гарантії якості , доставка, умови оплати та ін.).

Таблиця 5.18 – Визначення ключових переваг концепції потенційного товару

Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
Швидкість обробки даних	Швидка обробка вхідних даних	Алгоритми реалізовані найоптимальнішим способом та забезпечують найменшу затримку при обробці
Зручність застосування	Не потребує складного налаштування для впровадження у систему	Заощаджує час впровадження у систему за рахунок створеної архітектури
Точність обробки даних	Висока точність виявлення загрози	Висока вірогідність виявлення атаки за рахунок використання сучасних алгоритмів

Після формування маркетингової моделі товару слід відмітити, що проект буде захищено від копіювання за допомогою ноу-хау. Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 5.20). Аналіз проведено експертним методом.

Наступним кроком є визначення оптимальної системи збуту, в межах якого прийняте рішення (табл. 5.21).

Останньою складової маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 5.22).

Таблиця 5.19 – Опис трьох рівнів моделі товару

<table>
<tr><td>Рівні товарів</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>Товар за задумом</td><td colspan="3">Зручність та швидкість отримання практичного результату.</td></tr>
<tr><td>Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td rowspan="4">Товар із підкріпленням</td><td>1.функція для запобігання Brute force атаці
2. функція для запобігання мережевій DoS-атаці
3. функція для запобігання перехоплення мережевого трафіку</td><td></td><td></td></tr>
<tr><td colspan="3">Якість: надійний захист від мережевих загроз</td></tr>
<tr><td colspan="3">Пакування: відсутнє</td></tr>
<tr><td colspan="3">Марка: PlayProtection</td></tr>
<tr><td rowspan="2">Товар із підкріпленням</td><td colspan="3">До продажу: відсутнє</td></tr>
<tr><td colspan="3">Після продажу: персональна підтримка з можливістю розширення функціоналу під власні потреби</td></tr>
<tr><td colspan="4">Вихідний код закритий. На ідею зареєстровано патент.</td></tr>
</table>

Таблиця 5.20 – Визначення меж встановлення ціни

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
30000 грн	44000 грн	У всіх трьох груп достатній рівень доходів	Базова покупка 25000грн Подальша персональна підтримка в обслуговуванні 2500 грн

Результатом підрозділу стала ринкова (маркетингова) програма, що включає в себе концепції товару, збуту, просування та попередній аналіз можливостей ціноутворення, спирається на цінності та потреби потенційних клієнтів, конкурентні переваги ідеї, стан та динаміку ринкового середовища, в межах якого впроваджено проект, та відповідну обрану альтернативу ринкової поведінки.

Таблиця 5.21 – Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Цільові клієнти – компанії, які бажають впровадити у своїй роботі сучасні засоби, які забезпечать надійність та безпеку користувачів системи. Вони цікавляться існуючими рішеннями та інноваціями у сфері безпеки програмного забезпечення.	Встановлення контактів із споживачами і підтримання їх. Формування попиту і стимулювання збуту. Дослідницька робота зі збору маркетингової інформації. Забезпечення оборотного зв’язку зі споживачами.	Один (від виробника одразу споживачу)	Прямий канал збуту до споживача, мінімізувати збутові витрати розвиток маркетингового спілкування із споживачем

Таблиця 5.22 – Концепція маркетингових комунікацій

Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
Цільові клієнти – компанії, які бажають впровадити у своїй роботі сучасні засоби, які забезпечать надійність та безпеку користувачів системи. Вони цікавляться існуючими рішеннями та інноваціями у сфері безпеки програмного забезпечення.	Конференції, інтернет-конференції, семінари, огляд професійної літератури, інтернет, періодичні видання у різноманітних (профільних) галузях.	Позиція на основі порівняння фірми з товарами конкурентів; Відмінні особливості споживача	Створення репутації фірмі — виробнику чи посереднику; збільшення чистого прибутку та рентабельності фірми; збільшення потоків покупців та обсягів продажу; стабілізація обсягів продажу в період зменшення попиту та загального спаду ділової активності.	Забезпечте безпеку тим, хто вам довіряє!

Висновки за розділом

В даному розділі проведено аналіз програмного продукту у якості стартап-проекту. Можна зазначити, що у проекту є можливість комерціалізації, оскільки ринок потребує якісний продукт, що надає можливість інтелектуального редагування фотографій. На ринку наявна монополістична конкуренція, існує декілька фірм-конкурентів, але їх товар дещо відрізняється, тому вихід на ринок не буде легким і потребує грамотної стратегії виходу. Для впровадження ринкової реалізації проекту слід обрати альтернативу, яка передбачає розробку програмного продукту з подальшим розповсюдженням за певну плату. Можна сказати, що подальший розвиток проекту є доцільним, оскільки він знайде свою цільову аудиторію.

ВИСНОВКИ

У даній роботі проаналізовано загрози та способи захисту інформаційних систем, та наслідки, до яких може призвести недостатня увага до цього.

Розроблено метод для вибору оптимального вектору оптимізуючих перетворень. Запропонований алгоритм дозволяє обрати з множини оптимізуючих перетворень найкращу комбінацію для забезпечення найбільшого коефіцієнту захисту за умови обмеження у зменшенні швидкодії роботи системи навчання робототехніці.

Також описано структуру створеної веб-орієнтованої системи навчання робототехніці та її схема бази даних. Приведені результати тестування створеної веб-орієнтованої системи навчання робототехніці на вразливості атак, таких як, SQL-ін’єкція, Brute force, мережева DoS, перехоплення конфіденційних даних на стороні клієнта. Наведено блок-схеми алгоритмів, що були запропоновані для попередження даних видів атак.

Використовуючи запропонований метод вибору оптимального вектору оптимізуючих перетворень, були обрані перетворення та впроваджені у вихідному коді створеної системи навчання робототехніці.

Для комерціалізації даної роботи, розроблено стартап-проект.

ПЕРЕЛІК ПОСИЛАНЬ

1. Основи інформаційної безпеки. URL: https://www.intuit.ru/studies/courses/697/553/lecture/12442 (дата звернення 23.08.2019).
2. Основні поняття інформаційної безпеки. URL: https://studfiles.net/preview/6802020/ (дата звернення 26.08.2019).
3. Принципи інформаційної безпеки. URL: https://searchinform.ru/informatsionnaya-bezopasnost/osnovy-ib/ugrozy-informatsionnoj-bezopasnosti/ (дата звернення 02.09.2019).
4. Класифікація шкідливого програмного забезпечення. URL: https://studref.com/325279/informatika/klassifikatsiya_vredonosnogo_programmnogo_obespecheniya (дата звернення 06.09.2019).
5. Мережеві атаки, можливості та недоліки мережевих екранів. URL: https://www.bibliofond.ru/view.aspx?id=785336 (дата звернення 12.09.2019).
6. Мережеві атаки. URL: http://csaa.ru/ispolzovanie-specializirovannyh-programm (дата звернення 17.09.2019).
7. IP-спуфінг. URL: https://studopedia.org/11-30168.html (дата звернення 24.09.2019).
8. Написання захищеного коду. URL: http://inforsec.ru/technical-security/network-security/103-protected-code (дата звернення 26.09.2019).
9. IPsec. URL: https://lanmarket.ua/entsiklopediya/telekommunikatsionnye-tekhnologii/ipsec.html (дата звернення 01.10.2019).
10. TLS та SSL: Необхідний мінімум знань. URL: https://mnorin.com/tls-ssl-neobhodimy-j-minimum-znanij.html (дата звернення 04.10.2019).
11. Електронний цифровий підпис – що це, як зробити та отримати. URL:

https://kmb-chr.ru/sposoby-zarabotka/elektronnaya-tsifrovaya-podpis-chto-eto-kak-sdelat-i-poluchit-etsp.html (дата звернення 07.10.2019).

12. Загрози хмарних обчислень та методи їх захисту. URL: https://habr.com/ru/post/183168 (дата звернення 11.10.2019).
13. Інформаційна безпека САПР/PLM, що використовують хмарні технології. URL: https://cyberleninka.ru/article/n/informatsionnaya-bezopasnost-sapr-plm-primenyayuschih-oblachnye-tehnologii (дата звернення 15.10.2019).
14. Визначення та основні характеристики нечітких множин захисту. URL: http://nrsu.bstu.ru/chap21.html (дата звернення 17.10.2019).
15. Ус С.А., Коряшкіна Л.С. Моделі й методи прийняття рішень : навч. посіб. Дніпро : НТУ «ДП», 2018. 301 с.
16. Харниш, В. Правила прибыльных стартапов : как расти и зарабатывать деньги / В. Харниш ; пер. с англ. В. Хозинского. – Москва : Манн, Иванов и Фербер, 2012. – 279 с
17. Квашнин А. Как управлять портфелем технологий и интеллектуальной собственностью : серия методических материалов «Практические руководства для центров коммерциализации технологий» / под рук. П. Линдхольма, проект EuropeAid «Наука и коммерциализация технологий», 2006. – 60 с.

ДОДАТКИ

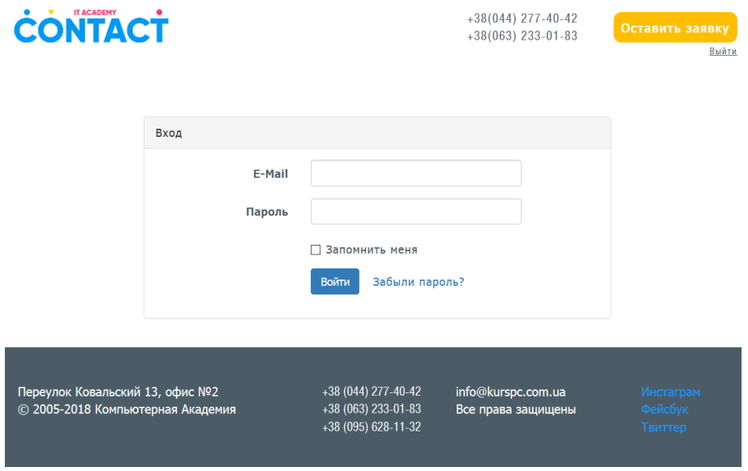
ДОДАТОК А

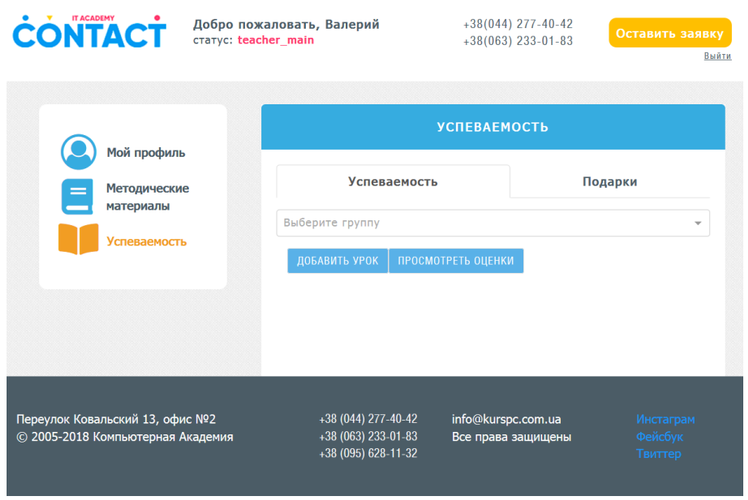
Плакати

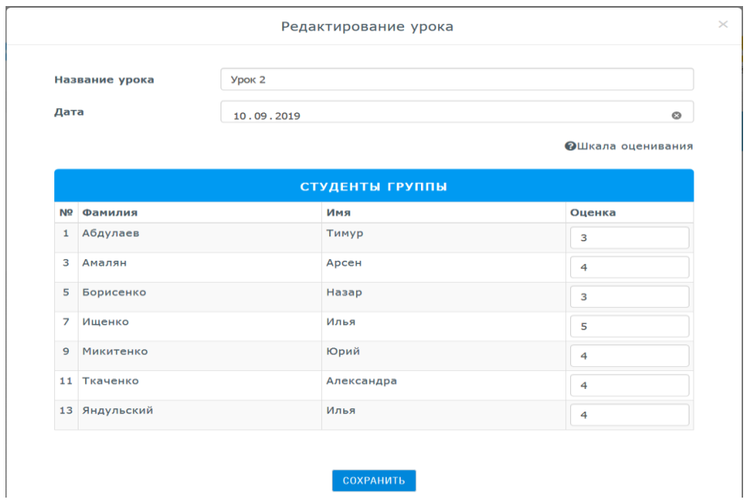
ДОДАТОК Б

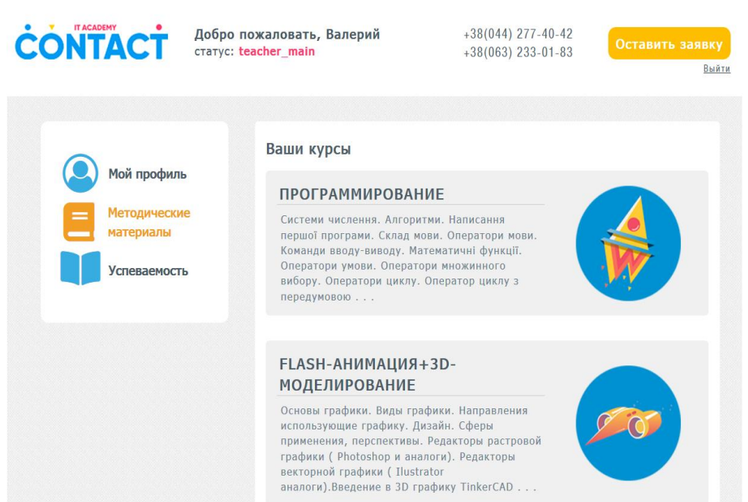
Перевірка на співпадіння

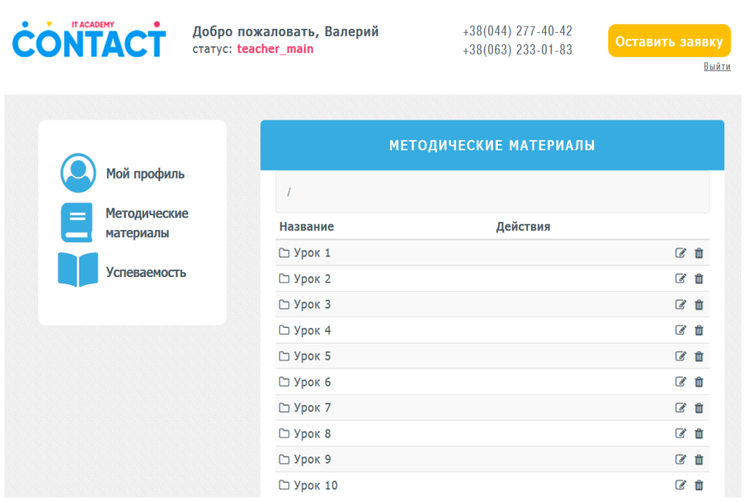
Екранні форми системи навчання робототехніці

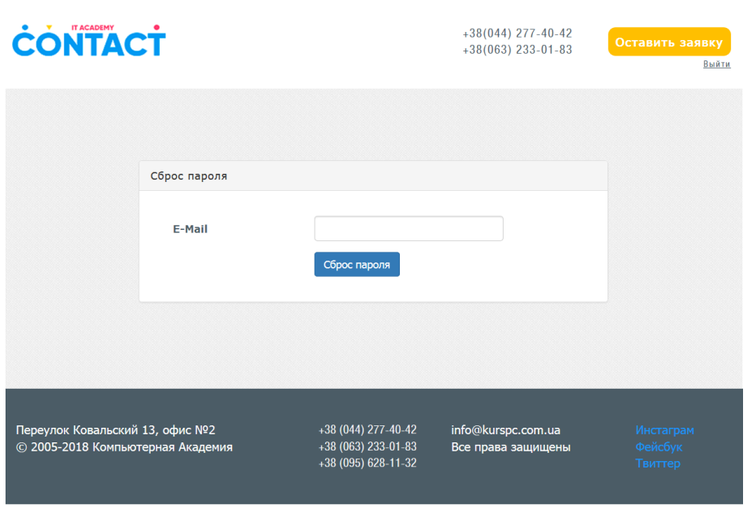








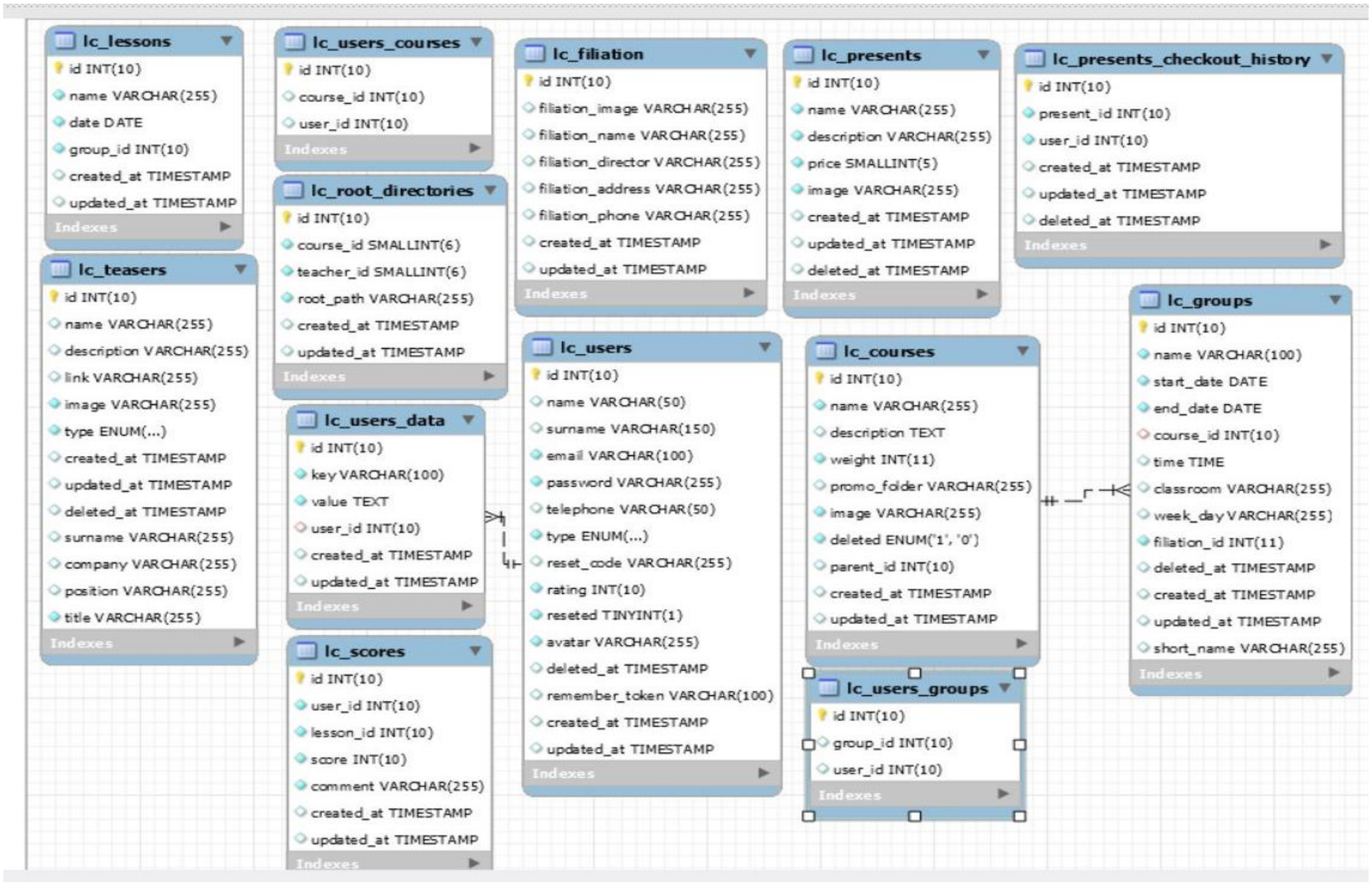




Демонстраційний плакат №1
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.

Структура бази даних



Демонстраційний плакат №2
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

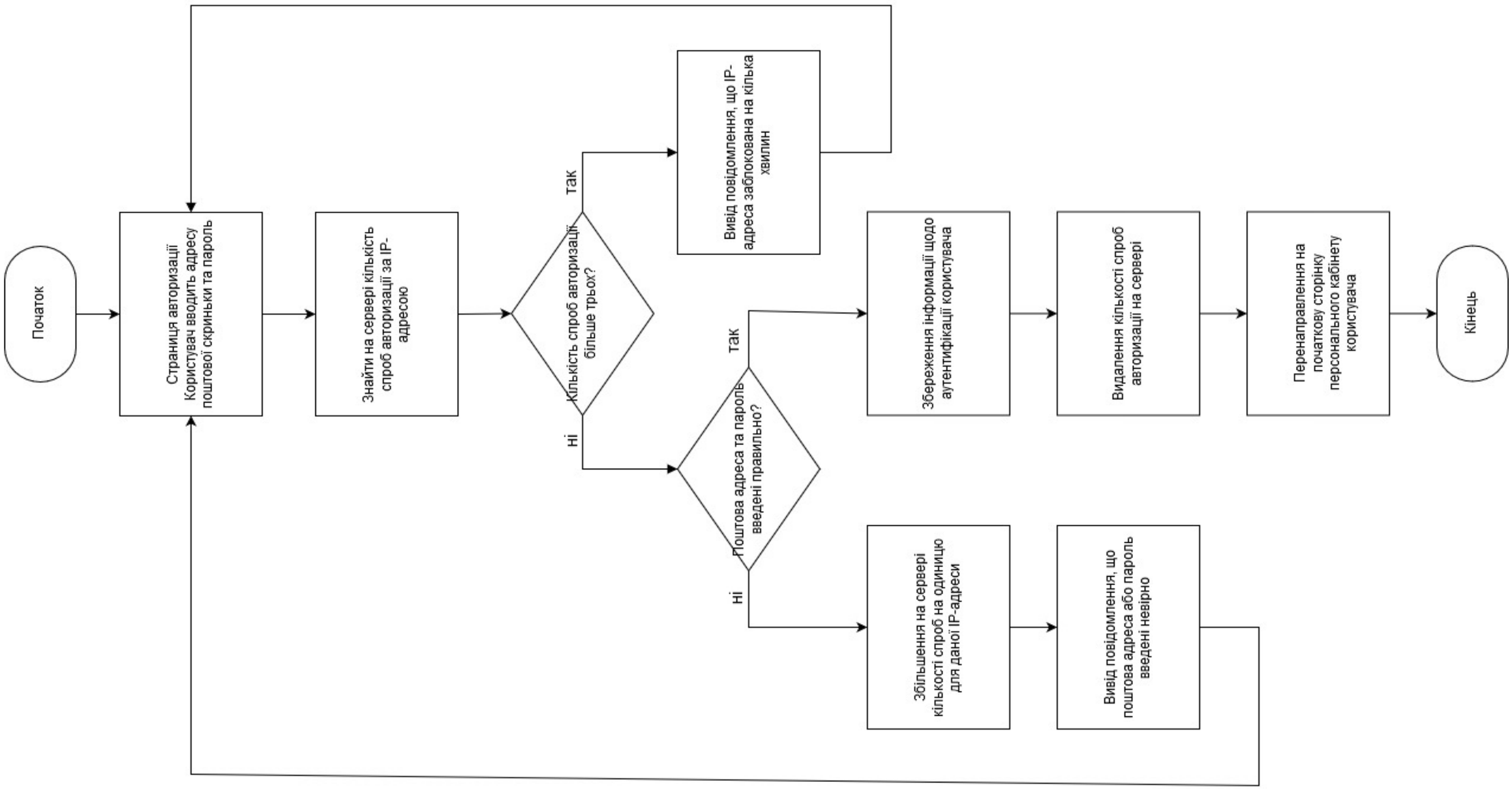
Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.

Схема системи навчання робототехніці

Демонстраційний плакат №3
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.

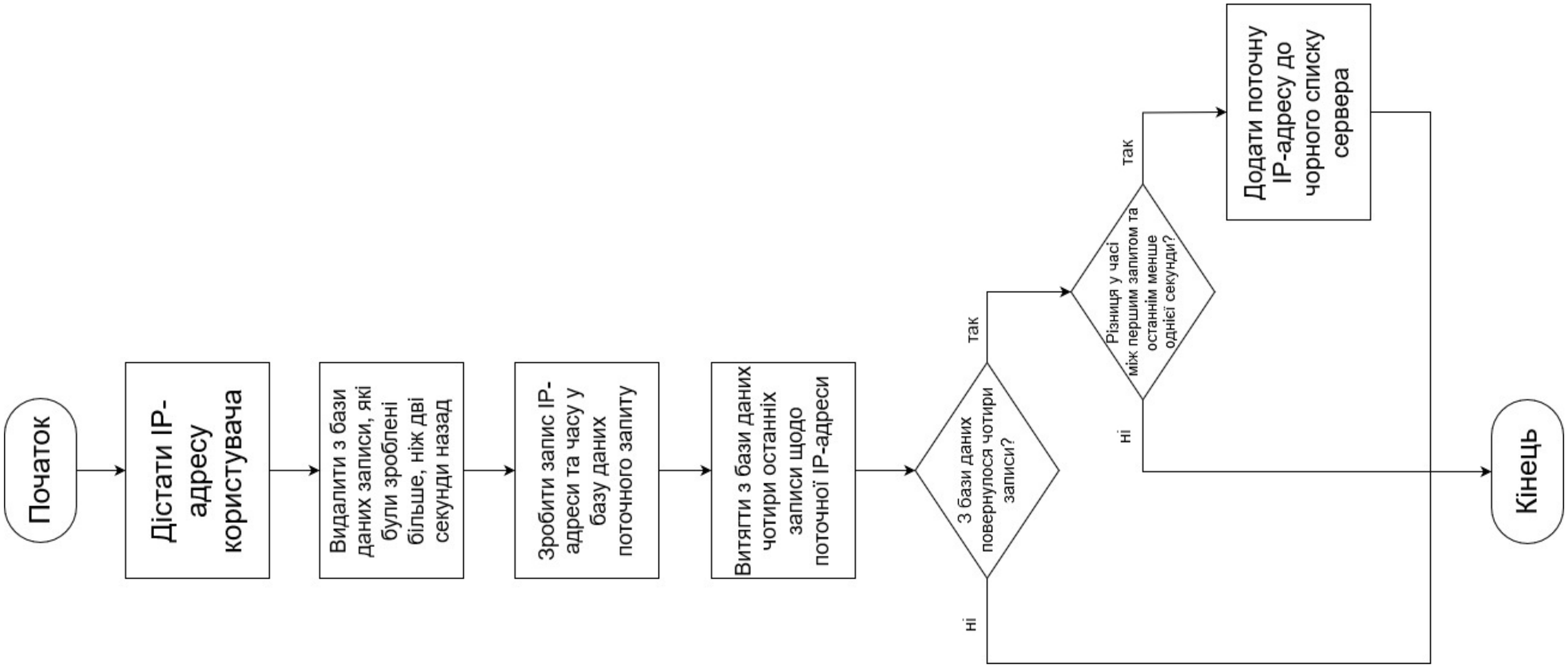
Алгоритм захисту від Bruto force атаки



Демонстраційний плакат №4
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.

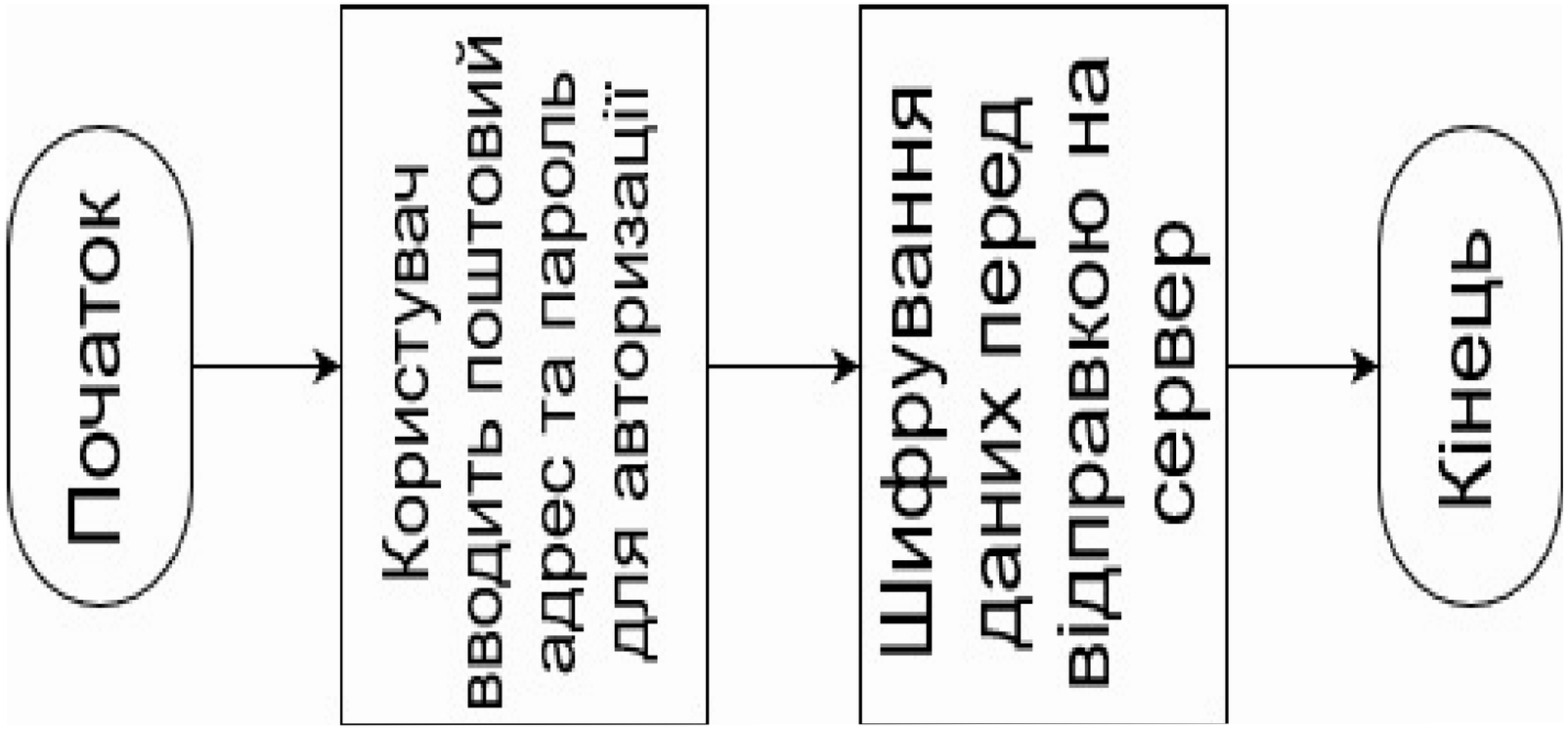
Алгоритм захисту від мережевої DoS-атаки



Демонстраційний плакат №5
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.

Алгоритм підвищення безпеки конфіденційних даних користувача на стороні клієнта



Демонстраційний плакат №6
до магістерської дисертації на тему
«Підвищення безпеки системи навчання робототехніці»

Виконав: студент гр. ІК-81мп Нікітін В. А.
Керівник: к.т.н., доцент Крилов Є. В.